



Universidade Nova de Lisboa
Faculdade de Ciências e Tecnologia

Departamento de Engenharia Electrotécnica e Computadores

Sistema Inteligente de Gestão da Manutenção e Suporte Operacional

Nelson dos Santos Silvério

Dissertação apresentada na Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa para obtenção do grau de Mestre em Engenharia Electrotécnica e Computadores

Orientador: José António Barata de Oliveira

Monte de Caparica

Fevereiro de 2010



Universidade Nova de Lisboa
Faculdade de Ciências e Tecnologia

Departamento de Engenharia Electrotécnica e Computadores

Intelligent Maintenance Management System and Operational Support

Nelson dos Santos Silvério

Dissertação apresentada na Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa para obtenção do grau de Mestre em Engenharia Electrotécnica e Computadores

Orientador: José António Barata de Oliveira

Monte de Caparica

Fevereiro de 2010

Resumo

Actualmente, a área da manutenção tem assumido um papel cada vez mais relevante na produtividade e organização de uma empresa em questões de sustentabilidade, desempenho, qualidade e ambiente, tendo vindo a ser fundamental no aumento da competitividade e nas mudanças estratégicas face às inconstantes condições do mercado mundial.

Graças às recentes evoluções tecnológicas e aos emergentes paradigmas, novas estratégias e métodos de manutenção são concebidos no contexto industrial. A aplicação da inteligência artificial e mecanismos de decisão nos sistemas de manutenção permitem detectar e prever a ocorrência de falhas nos processos de produção, evitando a quebra dos processos produtivos. No entanto, a falta de sistemas de suporte operacional evita o planeamento e uma maior capacidade de resposta dos operadores e técnicos nas operações de manutenção.

Esta tese apresenta a concepção de um novo sistema inteligente de gestão de manutenção baseado no paradigma dos agentes e que oferece mecanismos e serviços de suporte operacional.

Palavras-Chave:

Sistemas Inteligentes Distribuídos, Interoperabilidade, e-Manutenção, Suporte Operacional.

Abstract

Nowadays, the maintenance area has been playing an increasing role in the productivity and organization of a company on sustainability, business performance, quality and environment issues for increasing competitiveness and changing strategies in the unpredictable world market.

Thanks to recent technological developments and emerging paradigms, new strategies and methods of maintenance are designed in industrial context. The application of artificial intelligence and decision-making mechanisms in the maintenance systems helps to detect and predict, preventing failures and breakdowns of production processes. However, the lack of support systems avoids the planning and greater responsiveness of operators and technicians in maintenance.

This thesis presents the design of a new intelligent management maintenance system agent-based and provides mechanisms and operational support services.

Keyword:

Distributed Intelligent Systems, Interoperability, e-Maintenance, Operational Support.

Agradecimentos

Após esta longa e enriquecedora etapa, quero prestar os meus profundos agradecimentos e reconhecimentos aos meus pais, irmãos e avós por me terem proporcionado chegar aonde cheguei e ser aquilo que sou hoje. Outro pilar importante na minha vida é a minha namorada Milene que sempre me apoiou incondicionalmente nos desafios que enfrentei.

Gostaria de agradecer ao meu orientador, o Professor Dr. José Barata, pela confiança e atenção prestadas no decorrer desta tese e nas aulas das quais privilegiei com o mesmo.

Um obrigado especial ao meu colega e amigo, Mestre Luís Ribeiro, pela partilha e inspiração de ideias que permitiram enriquecer esta dissertação.

Não podia esquecer os meus colegas e companheiros de residência, Bruno Ferreira, Mauro Reis, José Oliveira e Erik Snelling nesta inesquecível aventura de cinco anos de convívio e partilha em longas horas de estudo.

Um muito obrigado aos meus parceiros Michel Rodrigues, António Amado, Amílcar Ferreira, Alexandre e Tiago Oliveira pela sua companhia e colaboração nos trabalhos de grupo.

Por fim, um agradecimento especial aos meus colegas de trabalho da PTSI, Adérito Ferreira, João Leite e colegas do projecto SGP e, também ao meu ex-manager da empresa

Agap2, André Lopes, pelas palavras de incentivo e atenção prestada na última fase de concretização desta dissertação.

Faculdade de Ciências e Tecnologia, Fevereiro de 2010

Nelson dos Santos Silvério

Índice

Resumo	i
Abstract.....	ii
Agradecimentos	iii
Índice	v
Índice de Figuras	viii
Índice de Tabelas	xi
Lista de Siglas.....	xii
1 Introdução	14
1.1 Motivação	14
1.2 Objectivos da Tese.....	18
1.3 Organização da Tese	19
2 Estado da Arte da Manutenção	20
2.1 Evolução e estratégias da Manutenção	20
2.2 <i>E-Maintenance</i> : Uma nova realidade.	25
3 Tecnologias de Suporte	30

3.1	XML como base de dados	30
3.2	Web Services	32
3.2.1	Definição	32
3.2.2	WSDL.....	32
3.2.3	SOAP	33
3.2.4	UDDI	33
3.3	Agentes	33
3.3.1	Definição e Características	33
3.3.2	Arquitecturas	35
3.3.3	Plataforma JADE.....	36
3.3.4	Plataforma WSIG	40
4	Arquitectura	44
4.1	Contextualização.....	44
4.2	Sistema de Gestão e Suporte Local de Manutenção	48
5	Implementação.....	54
5.1	Funcionalidades dos Agentes.....	54
5.2	Integração dos agentes JADE com Web Services	58
5.3	Base de conhecimento de suporte à Manutenção	61
5.4	Interfaces Gráficas	63
5.4.1	Consulta de Operadores de Manutenção	64
5.4.2	Consulta de Especificações de Equipamentos.....	67
5.4.3	Consulta de Mapeamento de Erros e Problemas	69

5.4.4	Consulta de Equipamentos registados no Sistema de Manutenção.....	71
5.5	Caso de Teste	74
5.5.1	Configuração da plataforma WSIG	74
5.5.2	Aplicação Cliente do Sistema Inteligente de Manutenção	77
5.5.2.1	Serviço de Registo de Equipamento.....	77
5.5.2.2	Serviço de Notificação de <i>Report</i> de Diagnóstico	80
5.5.2.3	Serviço de Notificação de <i>Report</i> de Prognóstico.....	83
5.5.2.4	Serviço de Notificação de <i>Report</i> de Maintenance	85
5.5.2.5	Serviço de Desregisto de Equipamento.....	86
6	Conclusões e Perspectivas	87
6.1	Conclusões	87
6.2	Perspectivas de Trabalho Futuro.....	89
7	Referências Bibliográficas.....	91
	Anexo I – Web Service de Manutenção	94
	Anexo II – Schemas.....	101

Índice de Figuras

Figura 1.1 - Objectivos da Engenharia da Manutenção (Dhillon 2006).....	15
Figura 2.2 - A manutenção numa perspectiva de tempo adaptado de (Han and Yang 2006) ..	21
Figura 2.3 - Curva da mortalidade - “ <i>bathtub curve</i> ”	22
Figura 2.4 - Relação entre e-maintenance, e-manufacturing e e-business (Koç, Ni et al. 2003)	27
Figura 3.5 - O Agente e o seu ambiente (Monostori, Váncza et al. 2006)	33
Figura 3.6 - Arquitecturas de agentes	35
Figura 3.7 - Arquitectura da plataforma multi-agente JADE (Bellifemine, Caire et al. 2003)	38
Figura 3.8 - Protocolos FIPA Request, FIPA Contract Net e FIPA Cancel-Meta (FIPA 2002)	39
Figura 3.9 - Interface Gráfica de gestão e controlo de Agentes JADE	40
Figura 3.10 - Integração da plataforma JADE com WSIG (Bellifemine, Caire et al. 2007)....	41
Figura 3.11 - Arquitectura da aplicação WSIG (JADE-Board 2008).....	42
Figura 4.12 - Interacção entre utilizadores, equipamentos e operadores com o sistema de gestão de manutenção.....	46
Figura 4.13 - Interacções entre módulos inteligentes e dispositivos	47
Figura 4.14 - Visão Global da Arquitectura.	48
Figura 4.15 - Módulo de Suporte Operacional de Manutenção.	51

Figura 4.16 - Módulo de Configuração de Equipas de Manutenção	52
Figura 5.17 - Publicação de um serviço de manutenção como Web Service.....	59
Figura 5.18 – Serviço de notificação do evento Diagnóstico (Classe JAVA).....	60
Figura 5.19 – Especificação do serviço de notificação de diagnóstico na Ontologia de Manutenção	61
Figura 5.20 - Interface Gráfica principal de Consulta à Base de Conhecimento de Suporte à Manutenção	64
Figura 5.21 - Lista de Operadores registados na plataforma de Manutenção	65
Figura 5.22 - Lista de competências de um operador de Manutenção	65
Figura 5.23 - Histórico de Operações de Manutenção de um operador	66
Figura 5.24 - Exemplo dos detalhes de uma operação de Manutenção.....	66
Figura 5.25 - Interface Gráfica para Consulta de Especificações de Equipamentos.....	67
Figura 5.26 - Janela de Propriedades Globais.	68
Figura 5.27 - Janela de Propriedades de Manutenção	69
Figura 5.28 - Interface Gráfica para consulta de Mapeamentos de Erros	70
Figura 5.29 - Consulta detalhada de uma solução de intervenção para um problema mapeado	71
Figura 5.30 - Interface Gráfica para Consulta de Equipamentos no Sistema de Manutenção .	72
Figura 5.31 - Detalhes de uma falha ocorrida no equipamento.....	73
Figura 5.32 - Consola Web da plataforma WSIG	76
Figura 5.33 - Propriedade para exposição de agentes serviço em Web Services.....	76
Figura 5.34 - Interface Cliente para Registo de Equipamentos no Sistema de Manutenção....	77
Figura 5.35 - Registo do Equipamento com sucesso.....	78
Figura 5.36 - Fluxograma de registo de equipamento no Sistema de Gestão de Manutenção .	78
Figura 5.37 - Interface de simulação de <i>reports</i>	79
Figura 5.38 - Interface para envio de <i>reports</i> de diagnóstico.....	80

Figura 5.39 - Excerto de e-mail enviado ao operador de manutenção	82
Figura 5.40 - Diagrama de sequência UML da notificação de report de diagnóstico	83
Figura 5.41 - Interface para envio de <i>reports</i> de prognóstico	84
Figura 5.42 - Diagrama de sequência UML da notificação de report de diagnóstico	84
Figura 5.43 - Interface para envio de <i>reports</i> de manutenção	85
Figura 5.44 - Diagrama de sequência UML da notificação de report de manutenção	86

Índice de Tabelas

Tabela 5.1 - Descrição das Funcionalidades dos Agentes.....	57
Tabela 5.2 - Serviços de comunicação do sistema inteligente de manutenção	58
Tabela 5.3 - Propriedades principais de configuração da plataforma WSIG	75

Lista de Siglas

AM	Alarming Module
AMS	Agent Management System
BD	Base de Dados
BDI	Belief-Desire-Intention
CBM	Condition-Based Maintenance
CM	Corrective Maintenance
CMMS	Computerised Maintenance Management Systems
DF	Directory Facilitator
ERP	Enterprise Resource Planning
FIPA	Foundation for Intelligent Physical Agents
ICT	Information and Communication Technologies
JADE	Java Agent DEvelopment Framework
JVM	Java Virtual Machine
IMS	Intelligent Maintenance Systems
MHD	Maintenance History Database
MMO	Maintenance Management Ontology

MOSM	Maintenance Operation Support Module
MTBF	Mean Time Between Failures
MTC	Maintenance Team Configuration
MTTF	Mean Time To Failure
POMAES	Problem-Oriented Multi-Agent-based E-service System
PM	Preventive Maintenance
RCM	Reliability-Centered Maintenance
SGML	Standard Generalized Markup Language
SOAP	Simple Object Access Protocol
SM	Scheduled Maintenance
TBM	Time-Based Maintenance
TPM	Total Productive Maintenance
TSM	Technical Support Manager
UDDI	Universal Description Discovery and Integration
UML	Unified Modeling Language
WSDL	Web Service Definition Language
WSIG	Web Service Integration Gateway
XML	eXtensible Markup Language
XSD	XML Schema Definition

1 Introdução

1.1 Motivação

Actualmente, face à instabilidade social e imprevisibilidade dos mercados financeiros, as organizações são pressionadas em adoptar novas medidas estratégicas de sustentabilidade e de gestão organizacional que possibilitam atingir novos requisitos na minimização de custos, no aumento da eficiência, na melhoria da qualidade de entrega de produtos e serviços, e garantir uma maior agilidade e flexibilidade nos seus processos de produção.

Desde da Revolução Industrial que a área da manutenção é vista como uma necessidade, no entanto dada à sua importância, a manutenção tem vindo a assumir um papel cada vez mais activo na gestão de recursos e de processos, nomeadamente na indústria da manufactura, tendo um maior envolvimento nas várias fases do produto tal como o desenho, concepção e fases de fim de vida.

Confrontado com esta nova realidade e juntamente com uma maior relevância nos aspectos ambientais, desempenho e eficiência energética nos processos de produção, nas últimas décadas o paradigma da manufactura tem-se direccionado para a realização de uma sociedade cada vez mais sustentável, isto é, o objectivo da manufactura não é somente produzir mais produtos de um modo eficiente, mas pelo contrário fornecer as funções

necessárias para a sociedade na minimização do consumo de recursos e energia (Takata, Kirnura et al. 2004).

A grande parte dos custos de uma organização é proveniente do departamento de manutenção, estimando-se entre cerca de 15% a 60% dos custos dos bens produzidos, dependendo do tipo de indústria. Como por exemplo, a indústria americana gasta mais de 200 bilhões de dólares cada ano, em equipamentos e operações de manutenção (Mobley 2002).

As intervenções desnecessárias e a falta de gestão de recursos e dos planos de produção são os reflexos de perdas consideráveis, numa economia mundial extremamente competitiva que se regista nos dias de hoje.

Dhillon reforça que é necessária uma efectiva gestão de recursos e práticas de manutenção, podendo influenciar positivamente em factores de sucesso tais como o preço, a rentabilidade, a qualidade, a distribuição, a segurança, e a velocidade de inovação (Dhillon 2006).

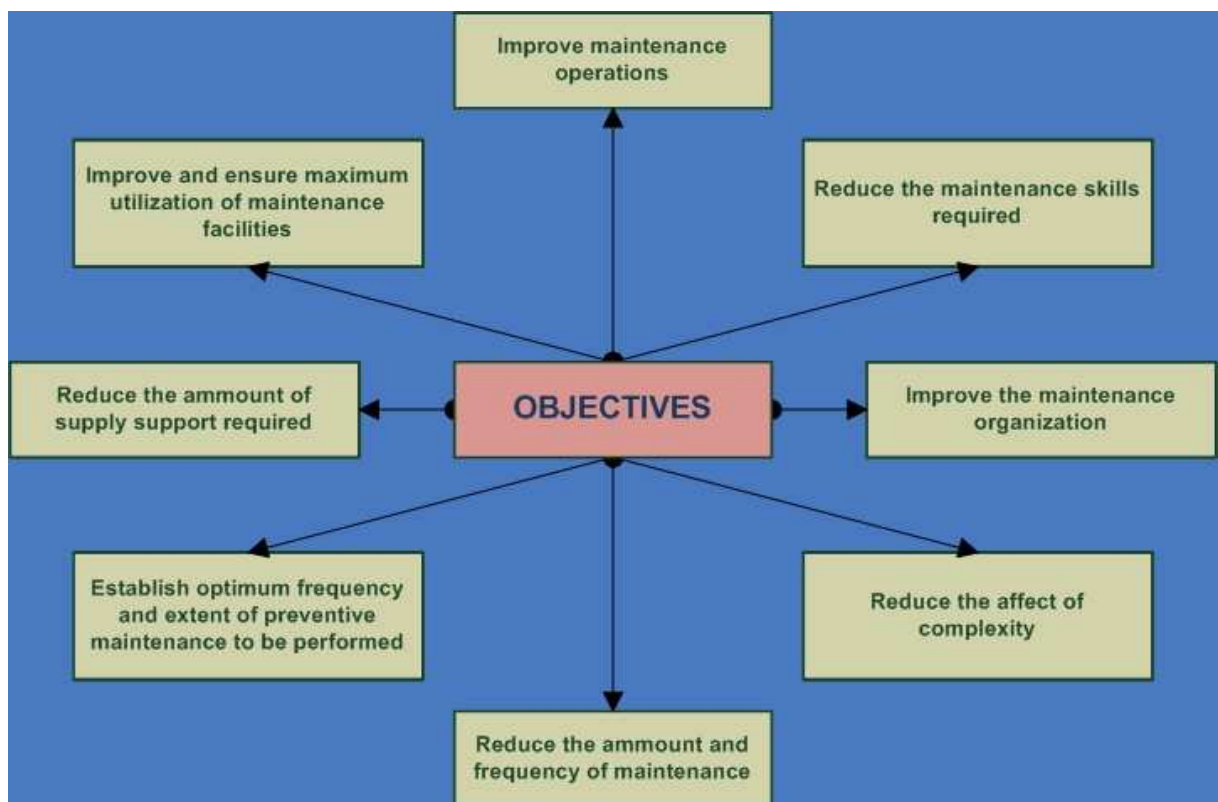


Figura 1.1 - Objectivos da Engenharia da Manutenção (Dhillon 2006)

Outro aspecto mencionado refere-se à engenharia de manutenção que está a tornar-se cada vez mais computadorizada e complexa, acabando por influenciar as novas actividades de manutenção dando ênfase às áreas de custo-efectividade, qualidade, segurança e factores humanos.

Na Figura 1.1 está ilustrado os principais objectivos da engenharia de manutenção.

Devido ao forte crescimento tecnológico e ao desenvolvimento da automação, a integração das tecnologias de informação e comunicação (ICT) tem proporcionado a exploração de novas técnicas e metodologias de manutenção. Estas modernas e eficientes manutenções implicam identificar, pelo menos, a causa primária de falhas nos componentes, reduzir as falhas de produção, eliminar os elevados custos de manutenção não programados, e melhorar a produtividade bem como a qualidade. Deste modo, as tradicionais práticas de manutenção devem consequentemente ser transformadas em metodologias proactivas face às novas considerações de sustentabilidade de uma organização (Benoit and Adolfo Crespo 2006).

Com a introdução de aplicações industriais, as estratégias de manutenção condicionada ou monitorizações on-line dos equipamentos tem sido integradas para prevenir possíveis degradações ou falhas de equipamentos (Garcia, Guyennet et al. 2004).

No entanto, o forte impulso das telecomunicações tem ajudado a controlar e a monitorizar remotamente plantas fabris geograficamente distribuídas, garantindo assim a manutenção e disponibilidade de cada sistema (Yang, Dai et al. 2007).

O conceito emergente de *e-Maintenance* (Marquez 2007) é o resultado da combinação destes vários factores já mencionados:

- O crescimento e evolução das *e-Technologies* na área da manutenção;

- A necessidade de integrar soluções de *Business Performance* através de serviços de monitorização, avaliação de desempenho, e práticas de manutenção nos sistemas de uma organização.

Apesar da emergência dos novos requisitos de sustentabilidade e da caracterização de objectivos da reengenharia da manutenção, a literatura pouco ênfase tem dado aos problemas de gestão dos recursos (físicos e humanos) e à falta de mecanismos adequados no suporte operacional e planeamento durante a execução das tarefas de manutenção, os quais são considerados os principais causadores de ineficazes medidas de intervenção e desadequadas paragens de produção que acabam por resultar em elevados custos e perdas numa organização.

Neste sentido, é importante ajustar e colmatar estas deficiências na gestão de uma organização aproveitando as novas soluções de comunicações e os recentes desenvolvimentos na área de *e-Maintenance*.

1.2 Objectivos da Tese

Baseado na evolução e novas tendências da manutenção, esta tese pretende abordar uma nova infra-estrutura inspirada no conceito de *e-Maintenance*, viabilizando e melhorando a gestão da manutenção e o suporte operacional em plantas fabris na área da manufactura.

Esta infra-estrutura foi projectada segundo os seguintes requisitos:

- Proporcionar a gestão de operadores e técnicos na formação de equipas especializadas para a ocorrência de intervenções de Manutenção;
- Disponibilizar serviços de comunicação e interacção com utilizadores remotos e/ou outros sistemas inteligentes através da exposição de interfaces genéricas de comunicação.
- Aproveitar a utilização de novas tecnologias de informação e técnicas de inteligência artificial, tais como sistemas multi-agentes, sistemas periciais e sistemas de apoio à decisão para facilitar a execução de operações através do aconselhamento especializado ao operador.
- Fornecer uma base conhecimento de informação útil à plataforma que possibilita identificar e listar possíveis causas e soluções para o suporte de operações de Manutenção (documentação funcional de equipamentos, histórico de manutenção dos recursos, etc.).

1.3 Organização da Tese

No âmbito dos objectivos apresentados, esta tese está organizada em 6 capítulos: Introdução, Estado da Arte da Manutenção, Tecnologias de Suporte, Arquitectura, Implementação e Conclusões e Perspectivas.

Este primeiro capítulo introdutório destaca as novas tendências de sustentabilidade das organizações, bem como o papel fundamental da manutenção nessa área. Por outro lado, *e-Maintenance* é destacado como principal tema de pesquisa e exploração desta tese.

O segundo capítulo apresenta o estado da arte da manutenção ao longo do último século, descrevendo a evolução das várias estratégias e tecnologias desenvolvidas. Também o conceito emergente de *E-Maintenance* é pormenorizado com destaque para a sua aplicação no mundo industrial.

No capítulo seguinte é feita uma descrição das tecnologias envolvidas e da plataforma de suporte na implementação do novo sistema de gestão e apoio à manutenção.

No quarto capítulo é abordado a arquitectura do sistema pretendido, detalhando a funcionalidade de cada componente constituinte.

No quinto capítulo é descrita a implementação do sistema desenvolvido e é apresentado o caso prático de utilização deste sistema.

Por último, é apresentado as conclusões da elaboração deste trabalho e discutidas as perspectivas de trabalho futuro.

2 Estado da Arte da Manutenção

Este capítulo apresenta uma perspectiva e visão histórica da evolução da área da Manutenção. O seu papel importante no contexto da sustentabilidade de uma empresa, tem despertado o interesse de investigadores na melhoria de técnicas e métodos apresentando novas estratégias de Manutenção consoante os critérios definidos. Assim, a influência do crescimento tecnológico tem proporcionado a emergência de uma nova área denominada *E-Maintenance* que tem revolucionado o conceito de manutenção na integração de serviços e recursos numa empresa.

2.1 Evolução e estratégias da Manutenção

Tradicionalmente, a área da manutenção foi claramente desprezada e descontextualizada na gestão industrial de processos até cerca de metade do século XX. Na Figura 2.2 é representado a evolução da manutenção numa perspectiva de tempo (Mobley 2002; Takata, Kirnura et al. 2004; Han and Yang 2006).

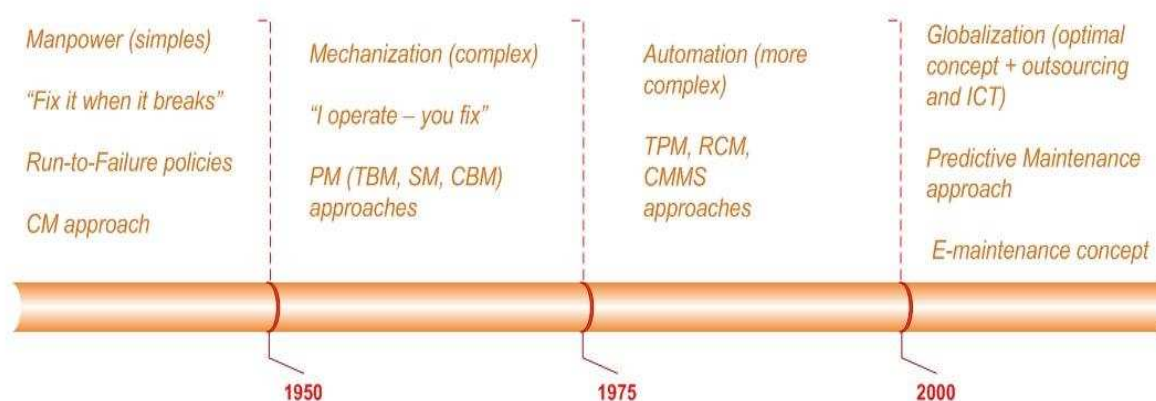


Figura 2.2 - A manutenção numa perspectiva de tempo adaptado de (Han and Yang 2006)

Verifica-se que até aos anos 50, a manutenção era entendida como uma actividade de reparação subjacente aos métodos e políticas tradicionais: *“run-to-failure”*. As tarefas de manutenção apenas eram desempenhadas quando detectada uma avaria ou falha nos recursos. Esta actividade era principalmente ineficiente e dispendiosa devido à inexistência de planos de manutenção com operações e interrupções não programadas de intervenção, levando a exaustão de funcionamento do sistema apresentando evidentes níveis de baixo desempenho.

Por outro lado, o tipo de actividade denominada por manutenção correctiva (***corrective maintenance*** - **CM**) tinha como objectivo a recuperação dos bens ou equipamentos para um específico estado quando detectado deficiências ou falhas, através de reparações ou manutenções planeadas e desempenhadas por operadores de manutenção, restabelecendo as condições óptimas de funcionamento do respectivo recurso ou sistema.

No entanto, o fraco desempenho das operações de manutenção na rentabilidade e eficácia dos processos de produção obrigou a reestruturar a organização da manutenção nas empresas com a criação de novas aproximações e técnicas para viabilizar a gestão da manutenção.

Neste contexto, de modo a resolver os longos períodos de inactividade nos processos de produção e avaria dos equipamentos, emergiu o conceito de manutenção preventiva

(*Preventive Maintenance* – **PM**) a partir da década de 50. Na literatura, a manutenção preventiva é classificada em três tipos de manutenção: manutenção baseada em intervalos de tempo, manutenção sistemática/programada e, por fim, a manutenção condicionada.

A manutenção em intervalos de tempo (*Time-Based Maintenance* - **TBM**) é baseada na curva da mortalidade ou mais conhecida como a curva de banheira, e permite verificar o aumento da taxa de falhas dos produtos após um certo período de operações (Figura 2.3).

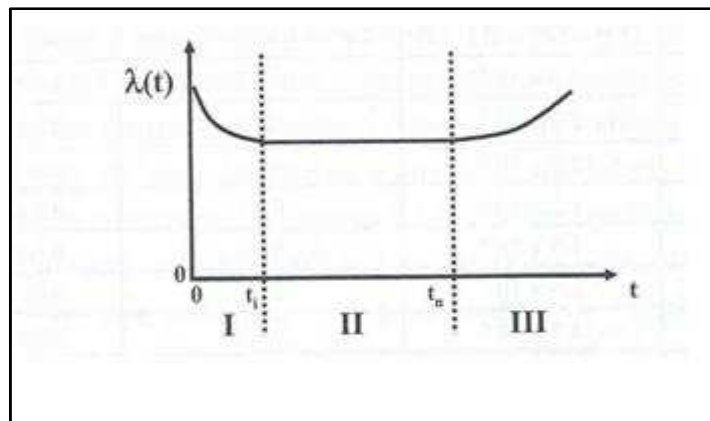


Figura 2.3 - Curva da mortalidade - “bathtub curve”

Contudo, esta relação não depende apenas do tempo decorrido mas também de condições de operação e de ambiente. Neste sentido, as rotinas de manutenção eram agendadas periodicamente no tempo mediante um conjunto de factores, evitando antecipadamente as falhas de um determinado recurso ou equipamento.

Mas as estratégias mais comuns são a manutenção preventiva sistemática/programada (*Scheduled Maintenance* - **SM**) e a manutenção preventiva condicionada (*Condition-Based Maintenance* - **CBM**).

A manutenção preventiva sistemática/programada estabelece rotinas de manutenção definidas em intervalos fixos de tempo durante o ciclo de vida do equipamento. Estes tempos são estabelecidos por recomendações do fabricante ou pela experiência dos operadores de manutenção. (Tang, Miller-Hooks et al. 2007)

Outros indicadores utilizados para a determinação dos tempos são baseados na estimação de medidas de manutenção de desempenho, nomeadamente disponibilidade, média dos tempos de bom funcionamento “*Mean Time Between Failures - MTBF*”, frequência de avarias, média dos tempos de falha “*Mean Time To Failure - MTTF*” e taxa de produtividade (Bukowski and Goble 2001; Mondro 2002; Amari 2006).

Por exemplo, as operações de manutenção (inspecções periódicas de equipamentos, lubrificações, reparações ou reconstruções) podem ser programadas baseado em estatísticas **MTTF**.

Assim, esta aproximação traz novas vantagens em que os custos de manutenção são predeterminados e as operações de manutenção são programadas de acordo com o ciclo produtivo.

No entanto, esta solução apresenta também desvantagens, apesar das paragens dos sistemas serem programadas, ainda que planeadas, estas têm um custo elevado.

Com a introdução da automatização dos sistemas nos anos 70, e com o avanço de técnicas de monitorização e diagnóstico, o conceito manutenção preventiva condicionada (*Condition-Based Maintenance - CBM*) emergiu (Yam, Tse et al. 2001).

Muito frequentemente, este tipo de manutenção é definido como uma manutenção inteligente. O acompanhamento do estado dos componentes intervenientes no ciclo produtivo e a detecção de sinais de degradação dos equipamentos (análise de parâmetros de desempenho, monitorização da vibração, termografia, análise do óleo, ferrografia, etc.) são os objectivos principais desta nova aproximação de manutenção.

Este tipo de aproximação permite enumerar algumas vantagens, tais como, um aumento de longevidade dos equipamentos, um maior controlo nos recursos, um menor custo de intervenção associado e um aumento de produtividade.

Oriundo da indústria japonesa, a manutenção produtiva total (***Total Productive Maintenance - TPM***) é adicionalmente, um outro conceito que para além de, combinar a eficiência total dos equipamentos e rentabilidade dos processos, procura melhorar o envolvimento activo de todos os que lidam com o processo e com o equipamento. Todo este enquadramento maximiza a disponibilidade dos recursos e pretende reduzir ou evitar a ocorrência de avarias (Tsang 2002).

Na década de 80, o termo “produtividade” foi substituído por “desempenho”, à medida que os critérios do paradigma de produtividade deixaram de satisfazer as partes interessadas.

Face a esta perspectiva, o conceito de Manutenção Centralizada na Fiabilidade (***Reliability-Centered Maintenance - RCM***) tornou-se uma realidade.

Inicialmente desenvolvido na indústria aeronáutica, o RCM é baseado na determinação de requisitos de bom funcionamento dos equipamentos e avaliação sistemática de avarias detectadas e efeitos causados na segurança, ambiente e na produção.

Esta metodologia pretende fortalecer os conhecimentos dos seus operadores de manutenção para enfrentar eventuais problemas e aplicar as adequadas técnicas de optimização e recuperação dos sistemas produtivos.

Consequentemente, esta metodologia apesar de ser optimizada e oferecer uma maior contextualização no plano operacional, esta acaba por tornar-se um processo demasiado moroso e extenso.

As características desta estratégia de manutenção podem ser encontradas em maior detalhe em (Moubray 1997).

Face a elaboração destes novos paradigmas e novas metodologias de manutenção, a manutenção tem sofrido várias alterações na estrutura de qualquer empresa e tem assumido um papel cada vez mais activo na gestão do ciclo de vida de um determinado produto ou bem (Takata, Kirnura et al. 2004).

Representando uma fatia substancial dos custos de uma organização (aproximadamente cerca de 15% a 60% dos custos totais de produção), a função da manutenção é visto como uma necessidade evidente na gestão dos níveis de produtividade dos recursos (equipamentos ou bens), qualidade, segurança e custos, mas também no planeamento, organização, implementação e controlo das actividades de manutenção.

Mais detalhes e informações acerca da dimensão estratégica da manutenção podem ser encontradas na seguinte documentação (Tsang 2002; Garg and Deshmukh 2006; Mobley, Higgins et al. 2008)

2.2 E-Maintenance: Uma nova realidade.

Na última década, o forte desenvolvimento das tecnologias tem contribuído para a mudança das estratégias de negócios na área da manufactura verificando-se uma transição nas questões de integração das empresas.

Face às constantes alterações das condições de mercado e uma maior atenção em questões sociais, regulamentações ambientais e segurança, as organizações tem direccionado as suas operações de manufactura para uma maior integração e completa globalização entre os serviços oferecidos, fornecedores e unidades de manufactura baseadas nas recentes tecnologias (*e-business*), de modo a manterem-se competitivas e responder às solicitações do mercado.

Neste contexto, uma nova estratégia de negócio emergiu – *e-manufacturing*. O conceito de *e-manufacturing* é definido segundo (Koç, Ni et al. 2003), como um sistema de transformação que viabiliza as operações de manufactura de modo a atingir níveis de performance óptimos (“zero avarias”), bem como, a sincronização com os sistemas de negócio (*e-business*) através de tecnologias *Wireless* e baseadas na Web.

Adicionalmente, *e-manufacturing* inclui a capacidade de monitorização dos recursos numa planta fabril, previsão da variação de qualidade do produto e perda de desempenho de qualquer equipamento para reprogramação da dinâmica de produção e operações de manutenção, e sincronização com os serviços prestados às empresas, para conseguir a perfeita integração entre a indústria de manufactura e sistemas empresariais de alto nível.

Desde do início de 2000, o termo de *e-maintenance* tem sido referenciado na literatura da manutenção como uma componente no conceito *e-manufacturing*. Contudo, a falta de consenso não tem permitido uma definição clara e consistente acerca deste termo.

Segundo (Benoit and Adolfo Crespo 2006), o conceito de *e-maintenance* está ligado a dois principais factores de emergência:

- O aparecimento de *e-tecnologies* permite o aumento da eficiência da manutenção, velocidade... e assim por diante, optimizando o fluxo de trabalho da manutenção (ou seja, *infotronics technologies*)(Lee 2003).
- A necessidade de integrar desempenho nos negócios, impõe na área da manutenção os seguintes requisitos: abertura, integração e colaboração com outros serviços de *e-enterprise* (novo modo de pensamento na manutenção).

Na Figura 2.4, pode-se verificar a integração entre *e-maintenance*, *e-manufacturing* e *e-business*, destacando a área de *e-maintenance* como um suporte estratégico no sucesso de integração entre estas áreas (Koç, Ni et al. 2003).

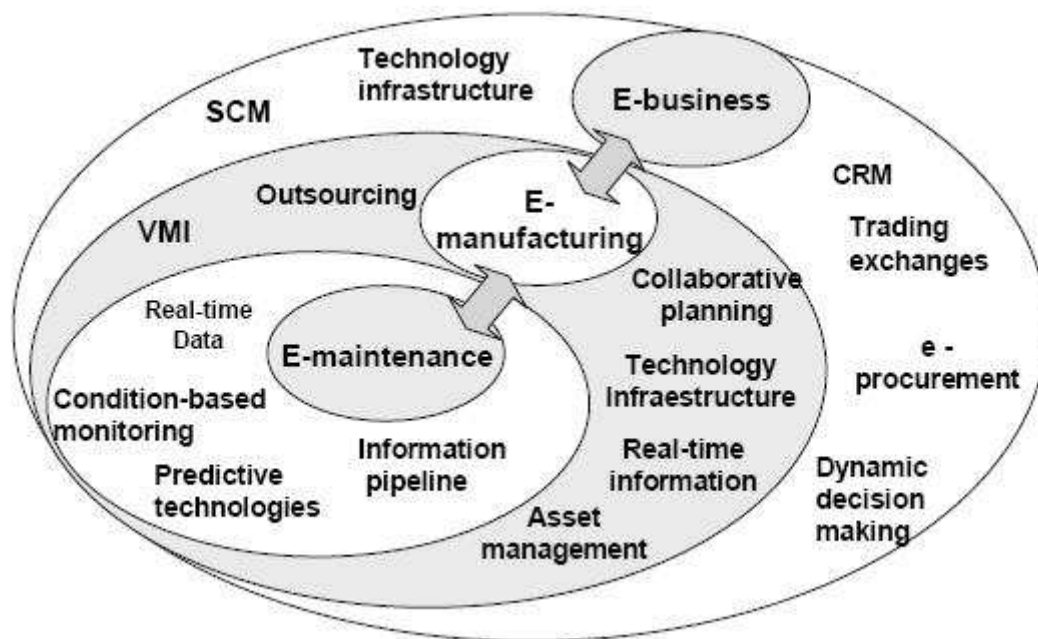


Figura 2.4 - Relação entre e-maintenance, e-manufacturing e e-business (Koç, Ni et al. 2003)

O conceito de *e-maintenance* tem conduzido à realização de diversas investigações e projectos um pouco por todo o Mundo.

Um dos mais activos centro de pesquisa em manutenção é o grupo *Intelligent Maintenance System (IMS)* (IMS 2007) que procura atingir melhor desempenho através de metodologias “zero avarias”.

O grupo IMS desenvolveu o projecto *Watchdog Agent™* (Jay, Jun et al. 2006) para implementar soluções **CBM**. Este sistema reúne dados a partir da monitorização de processos críticos através de sensores e relaciona-os juntamente com assinaturas para detecção de processos de degradação.

Além disso, o prognóstico é suportado pela aplicação de modelos estatísticos e de tendência sobre os processos de padrões enquanto o diagnóstico é baseado na memorização de relevantes assinaturas padrões de eventos passados.

Adicionalmente, o **Watchdog Agent™** está integrado com a plataforma **IMS Device-to-Business (D2B™)** que estabelece uma ligação entre o *shop floor* e *e-business* (Huang, Xi et al. 2005).

Em (Han and Yang 2006), uma plataforma de manutenção, baseado no conceito de centro de manutenção, é apresentado. Cada centro de manutenção é dividido em: **Fundamental research group** - pesquisa de tecnologias recentes e análise de simulação; **Enterprise group** – fornece fluxo de informação entre as empresas associadas; **Expert group** – concede suporte para a pesquisa de sistemas de manutenção e, melhora e otimiza os resultados de pesquisa, **Industry case history collection group** – reúne conhecimento através do ambiente industrial para manter uma base de dados e de modo a ser usada posteriormente em casos de raciocínio de diagnóstico, avaliação de riscos, centros de ajuda e suporte operacional, e estratégias de manutenção.

Além disso, a plataforma é baseada em centros de manutenção locais ao nível fabril que coordena as seguintes actividades: avaliação contínua do estado dos equipamentos, gestão de processos de reparação e manutenção, e representação e síntese de dados. Neste contexto, o sistema de manutenção local funciona como uma plataforma de teste para os resultados oriundos do centro de manutenção.

O projecto **problem-oriented multi-agent-based E-service system (POMAES)** proposto em (Yu, lung et al. 2003) explora a coordenação, cooperação e negociação em sistemas multi-agentes para solucionar problemas no contexto industrial revelando a noção de manutenção colaborativa.

Nesta abordagem, a implementação de um protótipo foi aplicado para controlar uma válvula de água. Nesta configuração três especialidades foram consideradas: **Production management expert** - controla o volume de água; **Maintenance expert** – otimiza o funcionamento da válvula e a disponibilidade da plataforma para minimizar os custos directos

e indirectos relacionados com a manutenção; **Control Expert** – garante o comportamento da instalação de acordo com as estratégias de produção.

Através de interacção um equilíbrio é atingido para maximizar a disponibilidade e manter os custos de manutenção controlados.

Outra plataforma de manutenção mencionada em (Szymanski, Bangemann et al. 2003; Bangemann, Thomesse et al. 2004; Bangemann, Rebeuf et al. 2006), engloba o projecto europeu **PROTEUS** que foca uma plataforma genérica de integração de sistemas, em vez do desenvolvimento de ferramentas de manutenção dedicadas. Do ponto de vista arquitectural, **PROTEUS** é constituído por diversas aplicações: *Web Portal*, *Enterprise Resource Planning (ERP)*, *Computerised Maintenance Management Systems (CMMS)*, *Knowledge management*, *e-Documentation server*, *Data Acquisition server* e *Platform core*. No entanto o principal objectivo desta plataforma é fornecer os meios necessários para promover a co-existência, cooperação e interoperabilidade destas aplicações proporcionando e oferecendo um serviço global e integrado no apoio às tarefas de Manutenção.

Por fim, uma completa descrição do conceito de *e-maintenance*, dos resultados obtidos e actuais pesquisas é fornecida em (Muller, Crespo Marquez et al. 2008).

3 Tecnologias de Suporte

O seguinte capítulo apresenta as tecnologias que serviram de suporte à implementação do projecto desta dissertação. Numa primeira abordagem, os aspectos de utilização de ficheiros **XML** como base de dados e respectivos documentos de caracterização e validação **XSD** são apresentados. De seguida, a importância da tecnologia dos *Web Services* no aumento da interoperabilidade entre sistemas é realçada. Por último, o paradigma dos agentes, o sistema multi-agente **JADE** e a plataforma de integração de *Web Service* – **WSIG** (Luck 2005; Monostori, Váncza et al. 2006) são referenciados.

3.1 XML como base de dados

Desenvolvido pelo consórcio W3C, *eXtensible Markup Language* (**XML**) é uma linguagem formal para formatação de documentos estruturados.

Segundo a especificação (Bray, Paoli et al. 1998), esta linguagem foi definida com os seguintes objectivos:

- Documentos **XML** devem ser utilizáveis na Internet.
- Os utilizadores devem perceber documentos **XML** directamente sem necessidade de serem processados e tratados.

- XML deve ser útil para uma vasta classe de aplicações.
- **XML** deve ser compatível com SGML – a linguagem precursora do **XML** para qual já existia conhecimento, experiência, ferramentas e documentos nesse formato.
- Devem conseguir facilmente escrever um programa para processar **XML**.
- As funcionalidades opcionais devem ser o mínimo, idealmente zero para tornar a linguagem simples e evitar problemas de aprendizagem e incompatibilidade.
- Os documentos devem ser directamente legíveis como texto.
- A especificação deve ser preparada rapidamente.
- **XML** deve ser definido formalmente e de forma concisa.
- Documentos **XML** devem ser fáceis de criar.
- **XML** deve evitar optimizações, evitando o aumento da complexidade sem aumentar a funcionalidade.

Para além dos documentos **XML**, a utilização de *Schemas* (ou chamado *Schema Definition Language*) é importante na descrição do formato de documentos XML e cujo ficheiros utilizam a extensão **.XSD** (Fallside and Walmsley 2004). A sua utilização é recomendada para estruturar os tipos de dados (simples ou complexos) e elementos, mas também para a validação da semântica do próprio documento **XML**.

Face à simplicidade e flexibilidade da linguagem, os documentos **XML** são utilizados em grande escala. Uma das suas principais utilizações é o armazenamento de dados em ficheiros **XML** como base de dados. Para um baixo volume de dados, os documentos **XML** são úteis face à sua manuseabilidade e flexibilidade de processamento.

No entanto, os documentos **XML** apresentam algumas desvantagens como o aumento de complexidade de processamento em grandes volumes de dados e ficheiros. A legibilidade dos documentos são afectados, enquanto o aumento de redundância de informação armazenada influencia o desempenho da performance no processamento dos documentos **XML**.

No aumento do volume e da complexidade da informação, a utilização de bases de dados relacionais são mais vantajosas pois permitem pesquisas mais eficientes em grandes quantidades de dados, melhor controlo de concorrência, recuperação de falhas e, para além disso, também suportam o armazenamento de documentos **XML**.

3.2 Web Services

3.2.1 Definição

Um *Web Service* é um componente de software implementado para suportar a interoperabilidade entre sistemas, e basicamente permite definir e descrever por intermédio de um **WSDL** (*Web Service Description Language*), uma interface de um serviço que pode ser publicado, descoberto e invocado, permitindo a integração entre aplicações de diferentes tecnologias (Booth, Haas et al. 2004).

3.2.2 WSDL

Web Service Description Language (**WSDL**) é uma linguagem que permite descrever em formato XML a interface de um serviço. Esta linguagem define basicamente o formato de uma mensagem, tipo de dados, protocolos de transporte e formatos de serialização de transporte (por exemplo **SOAP** e **XML RPC**) (Christensen, Curbera et al. 2001).

3.2.3 SOAP

Simple Object Access Protocol (SOAP) é um leve e extensível protocolo usado para troca de informação entre aplicações descentralizadas e ambientes distribuídos. O protocolo é definido em **XML** e é um mecanismo para invocar métodos sincronamente (Box, Ehnebuske et al. 2000).

3.2.4 UDDI

Universal Description Discovery and Integration (UDDI) é um repositório que permite o registo e acesso à descrição de **WSDL** através da publicação de uma interface pública (Clement, Hatelly et al. 2004).

3.3 Agentes

3.3.1 Definição e Características

O paradigma dos agentes é um conceito emergente do desenvolvimento de software inspirado nas teorias das tecnologias de informação, incluindo a inteligência artificial, a programação orientada a objectos e sistemas distribuídos.

No entanto, também é fonte de pesquisa em diversas áreas, nomeadamente na indústria da manufactura, telecomunicações, sistemas de transportes entre outros.



Figura 3.5 - O Agente e o seu ambiente (Monostori, Váncza et al. 2006)

Mas a definição de um agente não tem sido consensual nas várias áreas de investigação, mas permite contextualizar um agente como um sistema computacional, simples situado num ambiente dinâmico capaz de exibir autónomos e inteligentes comportamentos (Figura 3.5) com um conjunto de características e propriedades abaixo mencionado.

- **Autonomia** - Um agente tem a capacidade de controlar e tomar decisões e acções sem intervenção de humanos ou outros agentes.
- **Reactividade** – Um agente responde às imprevisíveis alterações de ambiente em que se encontra mediante a sua percepção.
- **Pró-actividade** – Um agente apenas não reage, mas exhibe um comportamento baseados nos seus objectivos tomando a iniciativa.
- **Social** – Um agente é capaz de cooperar e comunicar com humanos ou outros agentes de modo a cumprir as suas tarefas.
- **Adaptativo** – Um agente apresenta uma capacidade de aprendizagem, alterando de comportamento em cada ambiente ao longo do tempo.
- **Mobilidade** – Um agente tem a capacidade de se transferir numa rede de computadores, não alterando o seu estado interno.
- **Verídico** – Um agente não comunica deliberadamente informação falsa, mas fornece segurança.
- **Benevolência** – Um agente sempre tenta desempenhar o que lhe é solicitado.
- **Racional** – Um agente age de modo a conseguir os seus próprios objectivos, e nunca agir deliberadamente para prevenir que os seus objectivos sejam atingidos.

Face a estas características, os agentes estão preparados para interagirem em comunidades formando um ambiente multi-agente.

3.3.2 Arquitecturas

Ao longo dos anos, vários estilos de implementação foram definidos especificando cada sistema multi-agente com características distintas umas das outras mediante o seu papel de funcionamento.

Assim, as arquitecturas de agentes podem ser divididas em 3 principais grupos: deliberativo, reactiva e híbrida (Figura 3.6):

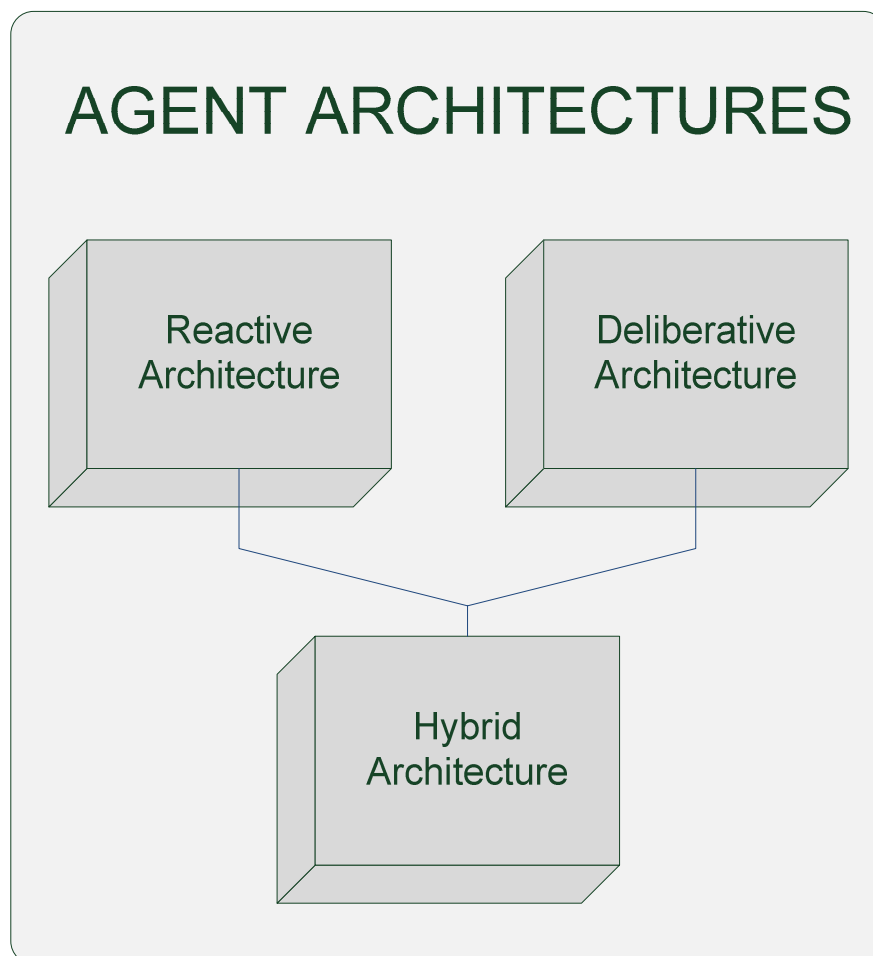


Figura 3.6 - Arquitecturas de agentes

A arquitectura deliberativa é caracterizada por agentes que tomam as suas acções ou comportamentos baseado num raciocínio lógico segundo uma representação abstracta e simbólica do mundo exterior. Uma das mais conhecidas arquitecturas deliberativas é a arquitectura **BDI** (*Belief-Desire-Intention*) em que o raciocínio lógico é constituído por atitudes mentais de crença, desejos e intenções. A vantagem deste tipo de arquitectura é que a representação do ambiente é simbólica e manipulada por mecanismos de raciocínio simples. No entanto, sendo uma representação simbólica dos conhecimentos não existem uma tradução directa, exacta e adequada do mundo exterior.

A arquitectura reactiva é caracterizada por agentes que agem segundo mecanismos de estímulos perante a resposta das suas acções num ambiente sem qualquer tipo de representação explícita. Este tipo de arquitectura é adequado para ambientes dinâmicos desempenhando as suas acções com maior simplicidade, rapidez e robustez. Por outro lado, não existindo uma representação externa do seu ambiente, a capacidade de resposta de um agente pode ser reduzida e insuficiente perante situações imprevisíveis e desconhecidas.

Por fim, a arquitectura híbrida resulta da conjugação das propriedades de ambas as arquitecturas anteriormente descritas, permitindo comportamentos de agentes deliberativos e reactivos, adoptando assim o melhor de cada aproximação (Luck, McBurney et al. 2005).

3.3.3 Plataforma JADE

JADE é uma plataforma de software totalmente desenvolvida em Java que simplifica a implementação de sistemas multi-agentes e aplicações distribuídas de acordo com os padrões da **FIPA** para agentes inteligentes .

As principais funcionalidades disponibilizadas pela plataforma **JADE** são apresentadas de seguida:

- Infra-estrutura de comunicação entre agentes com a definição de uma estrutura de mensagens, um eficiente mecanismo de transporte de mensagens assíncronas e serviços de codificação e decodificação de mensagens, seguindo um conjunto de especificações e protocolos **FIPA** que facilitam a interoperabilidade e interacção entre agentes;
- Directório de serviços de nomes (*naming service*), serviços de páginas amarelas (*yellow pages*) e páginas brancas (*white pages*) na plataforma de agentes.
- Gestão local ou remota do ciclo de vida de agentes (*Agent Management System - AMS*) fornecendo mecanismos de criação, suspensão, bloqueio e desbloqueio, migração, clonagem e término de agentes
- Suporte para a mobilidade de agentes facilitando a migração de agentes entre processos e máquinas. A migração de agentes é transparente, possibilitando mesmo a interacção entre agentes durante o processo de migração.
- Mecanismo de subscrição para agentes e aplicações externas na plataforma possibilitando a notificação de todos os eventos incluindo: eventos de troca de mensagens como eventos de ciclo de vida.
- Conjunto de interfaces gráficas e ferramentas de monitorização e *debugging* de execução de agentes.
- Mecanismos de segurança e suporte para a definição de ontologias e linguagens de conteúdo

A plataforma pode ser distribuída em diversas máquinas, no entanto, apenas uma aplicação **JAVA** e uma versão de **JVM** (*Java Virtual Machine*) é executada em cada máquina (Figura 3.7).

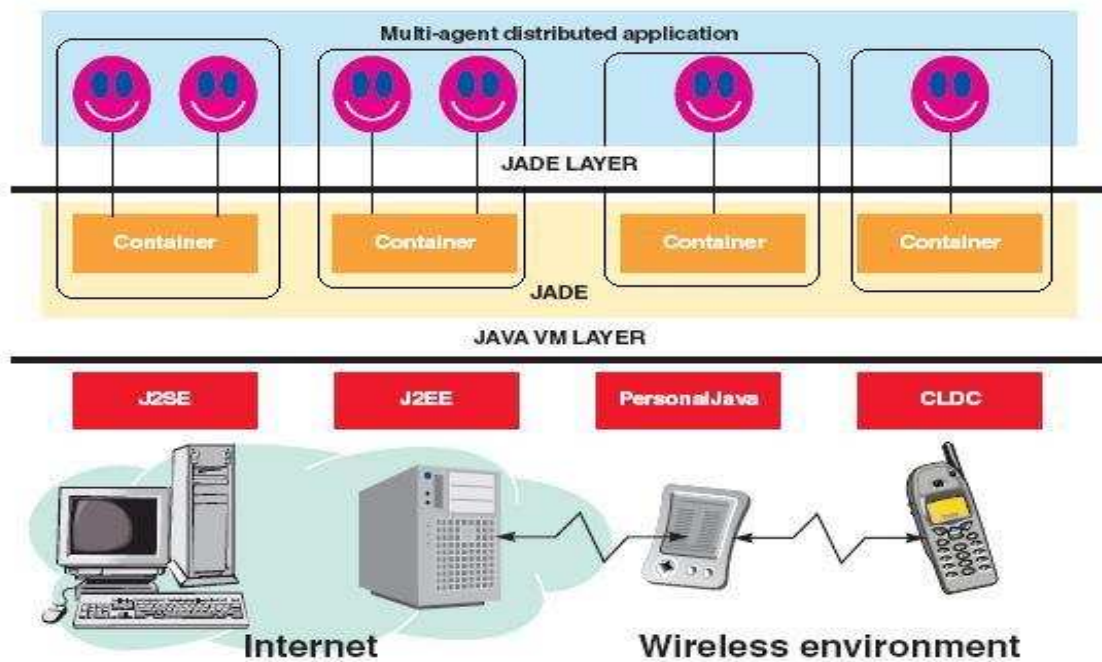


Figura 3.7 - Arquitetura da plataforma multi-agente JADE (Bellifemine, Caire et al. 2003)

A arquitetura de comunicação (Figura 3.8) oferece um flexível e eficiente sistema de mensagens, onde JADE cria e gere mensagens de entrada para cada agente. Para além disso, a arquitetura define um modo abstracto de pesquisa e comunicação entre agentes registados na plataforma, seguindo uma biblioteca de protocolos de interação – FIPA Request Interaction Protocol, FIPA Cancel-Meta Protocol, FIPA Contract Net Interaction Protocol, etc... (FIPA 2002; FIPA 2002).

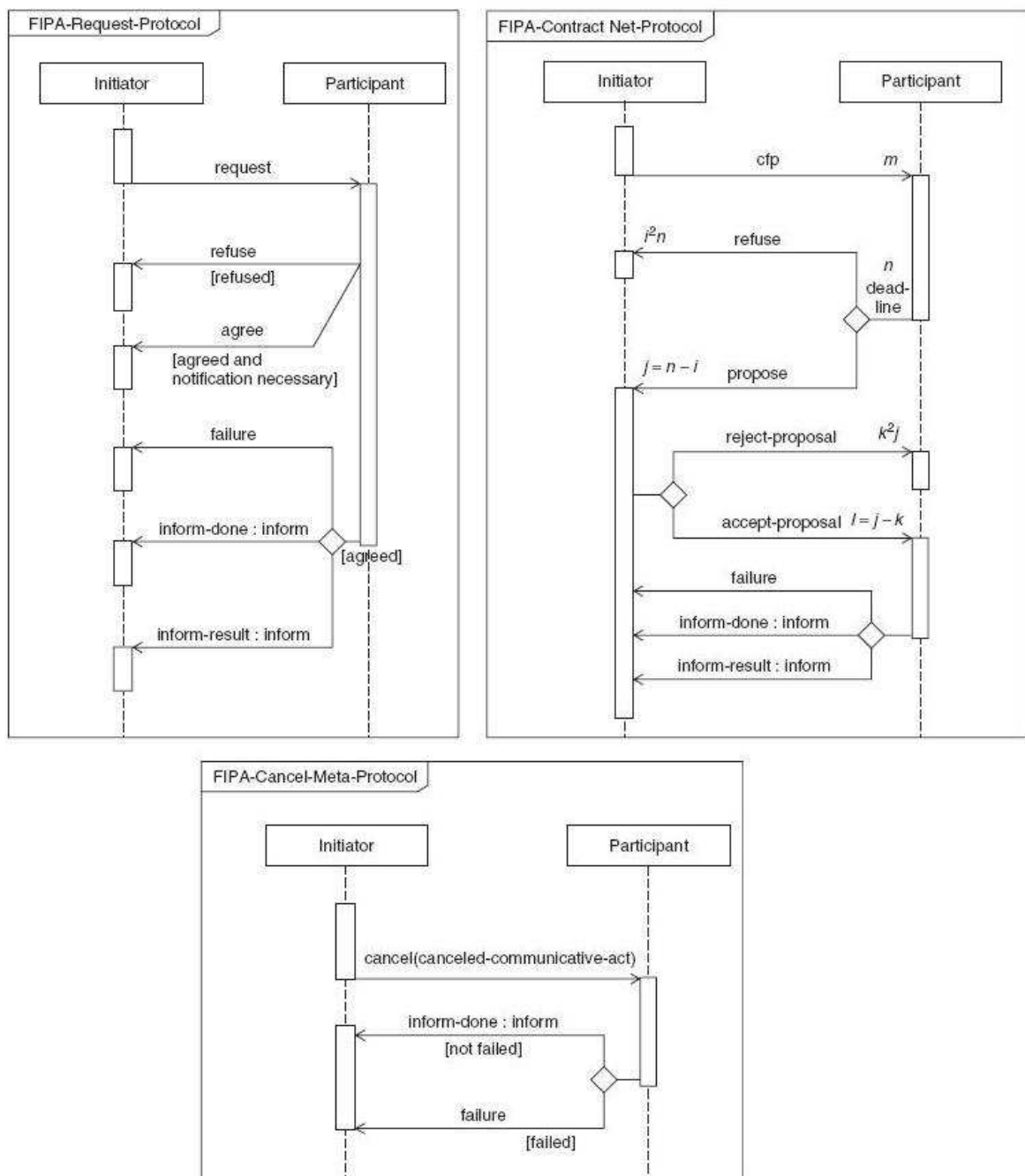


Figura 3.8 - Protocolos FIPA Request, FIPA Contract Net e FIPA Cancel-Meta (FIPA 2002)

Basicamente, os agentes são implementados como uma *thread* por agente, mas por vezes podem executar tarefas paralelas modelando um comportamento (*behaviour*).

Para gestão, monitorização e controlo remoto de cada agente registado, a plataforma de agentes **JADE** fornece uma interface gráfica (Figura 3.9) que permite operar um conjunto de funcionalidades, por exemplo, controlar remotamente o arranque ou paragem de um determinado agente numa determinada máquina, e também visualizar remotamente os

diagramas de interacção entre agentes de acordo com os protocolos FIPA anteriormente apresentados.

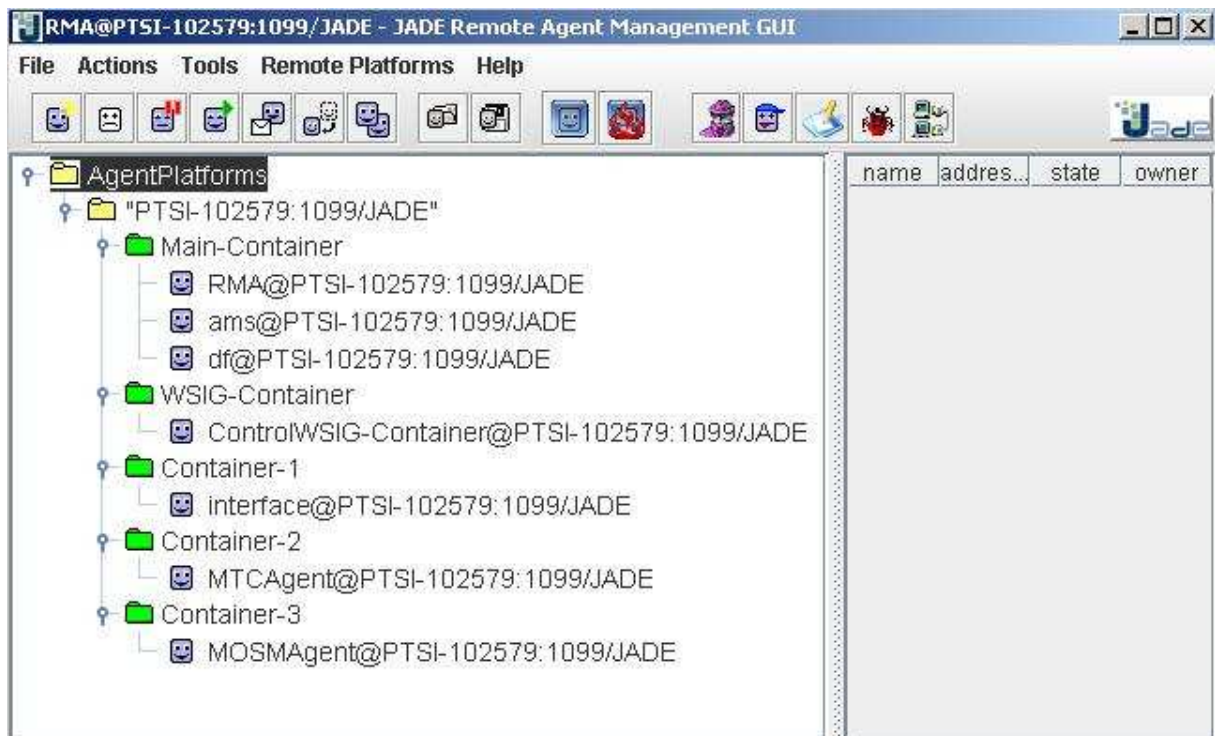


Figura 3.9 - Interface Gráfica de gestão e controlo de Agentes JADE

3.3.4 Plataforma WSIG

A plataforma *Web Service Integration Gateway* (**WSIG**) é um utilitário complementar à plataforma de sistemas multi-agentes **JADE**, que permite expor os serviços dos agentes registados no **DF** em **Web Services** (Figura 3.11). Este utilitário suporta os padrões dos **Web Services** constituídos pela descrição de serviços em **WSDL**, mensagens de transporte **SOAP** e um repositório **UDDI** para publicação de **Web Services**.

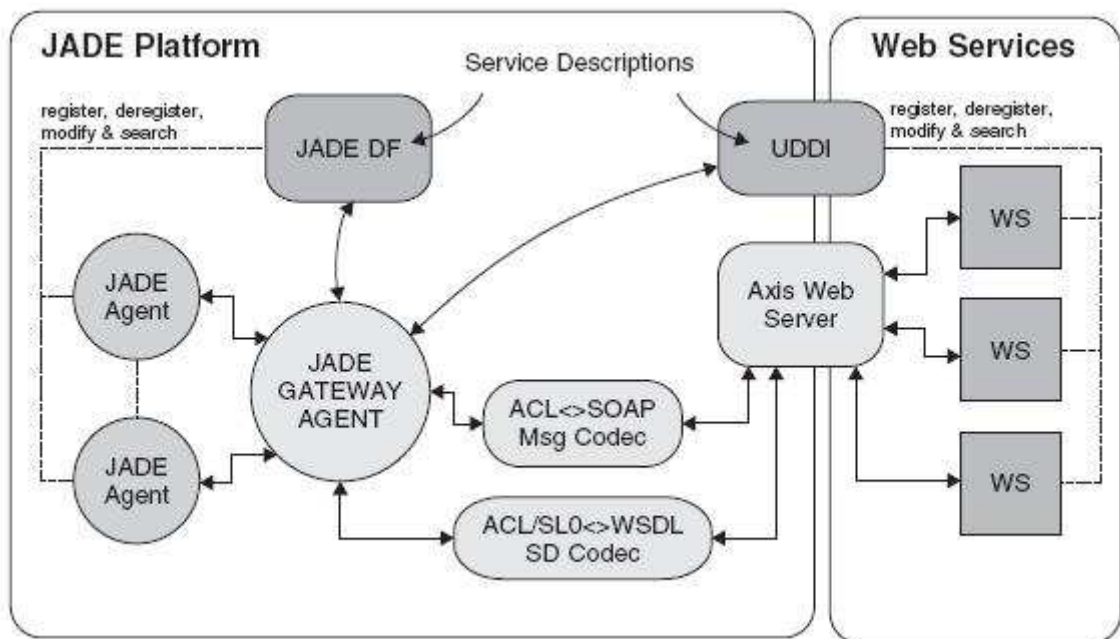


Figura 3.10 - Integração da plataforma JADE com WSIG (Bellifemine, Caire et al. 2007)

A aplicação **WSIG** é composta por dois principais elementos (Figura 3.11):

- Servlet **WSIG**
- Agente **WSIG**

O *servlet* **WSIG** é a interface responsável por:

- Processar as mensagens **HTTP/SOAP**.
- Extrair a mensagem **SOAP**.
- Preparar a correspondente acção agente e encaminhar para o agente **WSIG**.
- Converter o resultado da acção numa mensagem **SOAP**.
- Preparar a resposta **HTTP/SOAP** para ser devolvida ao cliente.

Por outro lado, o agente **WSIG** é a ligação entre a componente Web e os agentes, sendo responsável por:

- Encaminhar as acções agente recebidas a partir do *servlet* **WSIG** para os agentes capazes de processá-las e devolver o resultado.
- Subscrever no **DF** a recepção de notificações sobre registo/desregisto de agentes.
- Criar o **WSDL** correspondente para cada serviço agente registado no **DF** e publicar o serviço no **UDDI**, se necessário.

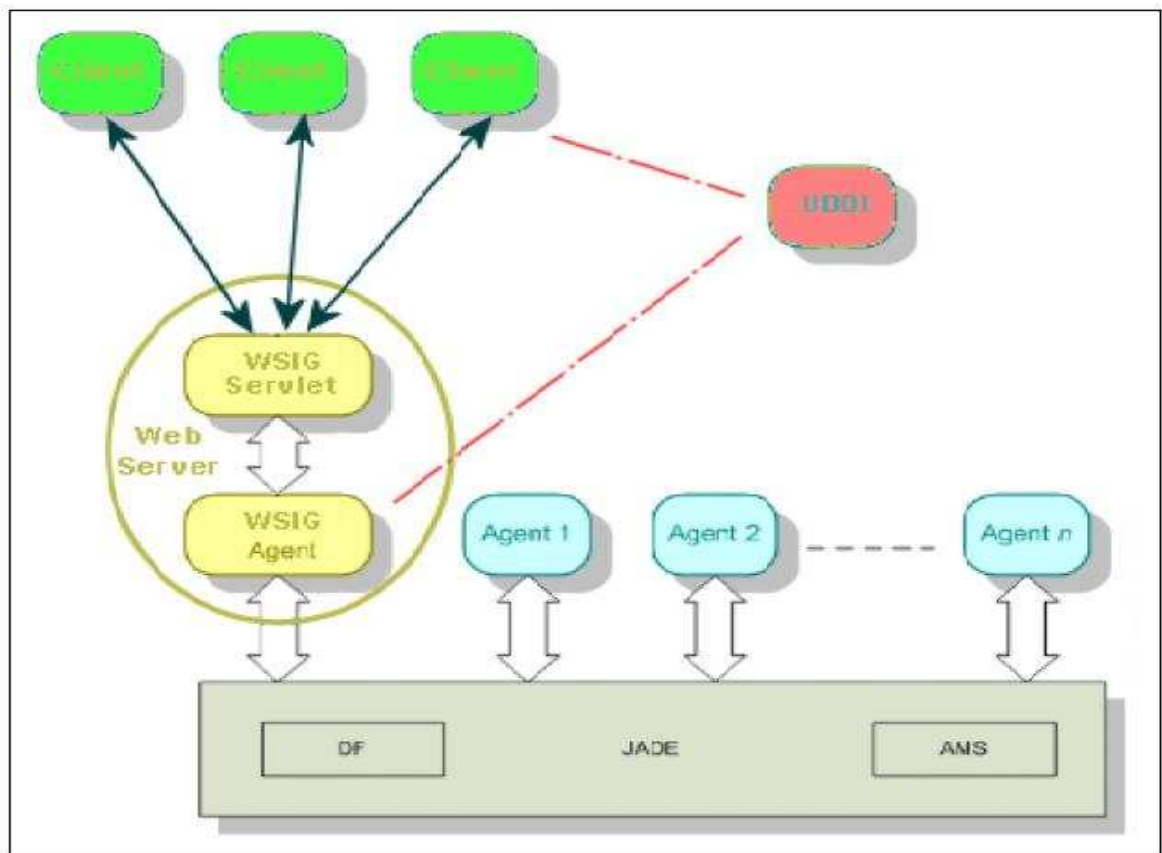


Figura 3.11 - Arquitectura da aplicação WSIG (JADE-Board 2008)

Na aplicação *web* **WSIG**, dois principais processos estão continuamente activos:

- O processo responsável pela intercepção de registos/desregistos no **DF** e pela conversão destes em WSDL desempenhado pelo agente **WSIG**.

- Por último, o processo responsável pelo tratamento dos pedidos de **Web Services** e pelo disparo das correspondentes acções agente, suportado juntamente entre o *servlet* **WSIG** e o agente **WSIG**.

Do ponto de vista funcional, esta integração da plataforma **JADE** e **WSIG** permite aumentar o grau de interoperabilidade entre sistemas, disponibilizando serviços que podem ser descobertos e invocados por outras entidades para além da tecnologia de agentes e plataforma não **JAVA**.

4 Arquitectura

Este capítulo detalha o objectivo principal desta tese descrevendo uma arquitectura de um sistema inteligente de apoio à gestão e suporte local de manutenção. Na primeira secção é abordado a contextualização do sistema desenvolvido. Na segunda secção é detalhada a arquitectura com a descrição de cada módulo do sistema e principais interacções intervenientes.

4.1 Contextualização

Como abordado no capítulo 2, a evolução da área manutenção tem direccionado os seus esforços na investigação de novos procedimentos, técnicas e processos de manutenção, e o desenvolvimento de sistemas que permitem melhorar a gestão da manutenção numa organização.

Neste contexto, suportado pelo conceito de *e-maintenance* é apresentado um novo sistema inteligente de apoio à gestão e suporte local de manutenção já referenciado em (Ribeiro, Barata et al. 2009).

O sistema pretende dar suporte na área da manutenção de equipamentos e de toda a envolvimento necessária para garantir uma manutenção evolutiva eficiente e reactiva.

Embora o foco desta tese seja a manutenção, as funcionalidades de auto-monitorização (*self-monitoring*) e auto-diagnóstico (*self-diagnosis*) não são explicitamente detalhadas na arquitectura, uma vez que é confiado que estes mecanismos já são disponibilizados pelo sistema.

Portanto o sistema oferece essencialmente mecanismos para gestão de recursos de manutenção (funcionários, técnicos e operadores de manutenção, etc.) ao longo dos processos de produção e durante o funcionamento dos diversos equipamentos intervenientes, formando equipas personalizadas e especializadas para as intervenções e operações de manutenção necessárias.

Na detecção de um problema num dos equipamentos previamente registado, o sistema de manutenção garante a colecção e entrega de toda a informação necessária para a criação e formação de uma equipa.

Esta informação inclui documentação, manuais e recomendações do fabricante dos equipamentos, o histórico das intervenções e operações de manutenção, a informação detalhada das competências e do histórico dos operadores de manutenção formando uma base de conhecimento de apoio ao Sistema de Manutenção.

Para além desta componente de gestão, o sistema inteligente permite acompanhar as intervenções durante a ocorrência de operações de manutenção dando suporte local, disponibilizando informação adicional para a resolução de avarias ou anomalias detectadas.

Adicionalmente, a plataforma permite a interacção de utilizadores do sistema, equipamentos e o estabelecimento de sessões remotas para operadores remotos com o sistema (Figura 4.12).

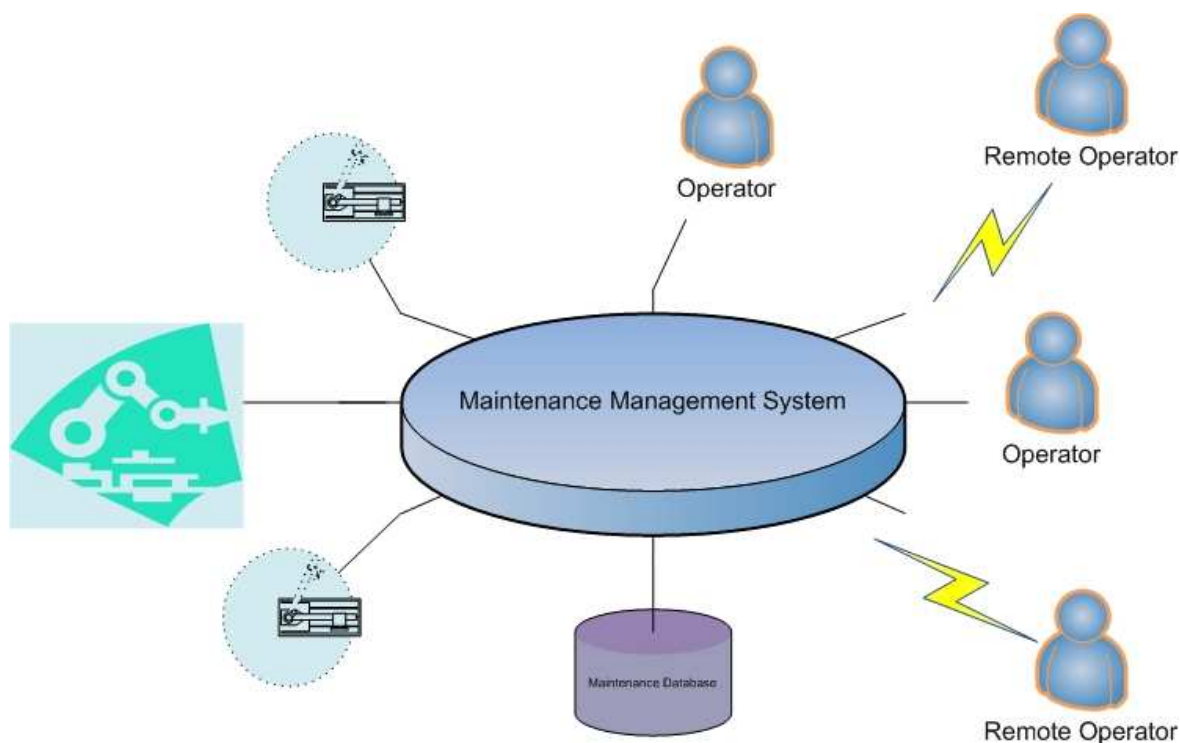


Figura 4.12 - Interação entre utilizadores, equipamentos e operadores com o sistema de gestão de manutenção.

Para a arquitectura do sistema foram definidos os seguintes requisitos:

- **Interoperabilidade (*Interoperability*)** – o sistema permite uma comunicação transparente com outros actores ou sistemas através da definição de interfaces genéricas de comunicação;
- **Modularidade (*Modularity*)** – o sistema é modular, evidenciando a separação de funcionalidades do sistema e permitir que uma alteração de uma das partes do sistema não afecte o funcionamento de todo o sistema;
- **Dinamismo (*Dynamic*)** – o sistema oferece um mecanismo de interação entre cada parte do sistema permitindo a adição ou remoção de funcionalidades ao sistema;
- **Flexibilidade (*Flexibility*)** – o sistema proporciona a habilidade de participação de diversos actores através de mecanismos de registo e desregisto no sistema;

- **Raciocínio (*Reasoning*)** – o sistema permite a interpretação de informação para previsão ou detecção de falhas através de mecanismos de *reasoning* (redes neuronais, modelos *fuzzy*, etc...);
- **Sistema Distribuído (*Distributed System*)** – o sistema opera de forma independente e autónoma, podendo cada módulo ser executado em diferentes máquinas ou qualquer ponto da rede partilhando um mecanismo de comunicação transparente.

Com base nas características mencionadas, o sistema foi projectado para apresentar uma componente infra-estrutural baseada na disponibilização de serviços de subscrição, serviços de alarmes e notificações, e mecanismos de auto-raciocínio (inteligência).

Na Figura 4.13 é ilustrado as principais interacções entre os vários módulos inteligentes que constituem o sistema e as interfaces genéricas de comunicação e de controlo.

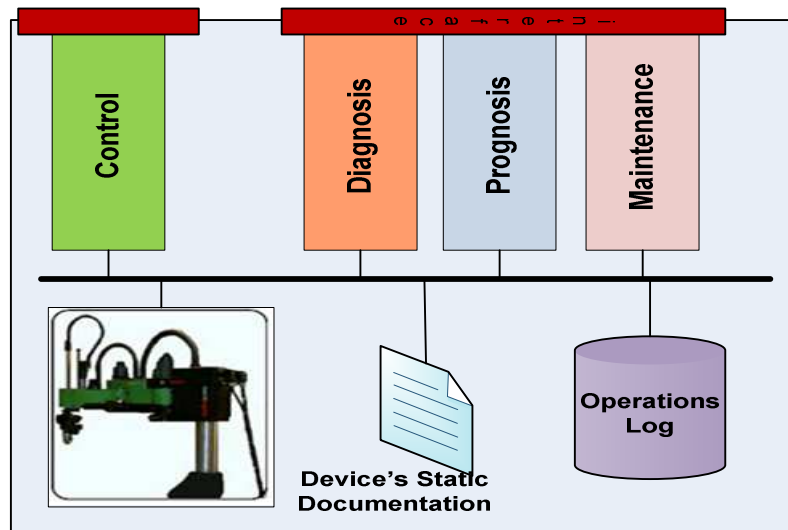


Figura 4.13 - Interacções entre módulos inteligentes e dispositivos

Para integrar o sistema, cada equipamento ou dispositivo efectua o seu registo através do serviço de subscrição de manutenção. Após o registo efectuado, cada equipamento pode notificar a plataforma acerca de eventos relevantes (manutenção, diagnóstico, prognóstico) invocando os serviços disponibilizados reportando dados sempre que necessário. O registo é

mantido através de uma sinalização periódica de controlo, informando a presença do dispositivo no sistema.

É importante verificar que a disponibilização de interfaces de comunicação permite facilitar e maximizar a integração e interligação (mecanismos *Plug and Play*) entre cada sistema.

4.2 Sistema de Gestão e Suporte Local de Manutenção

A arquitectura proposta para o sistema de gestão e suporte local de manutenção é ilustrada na Figura 4.14.

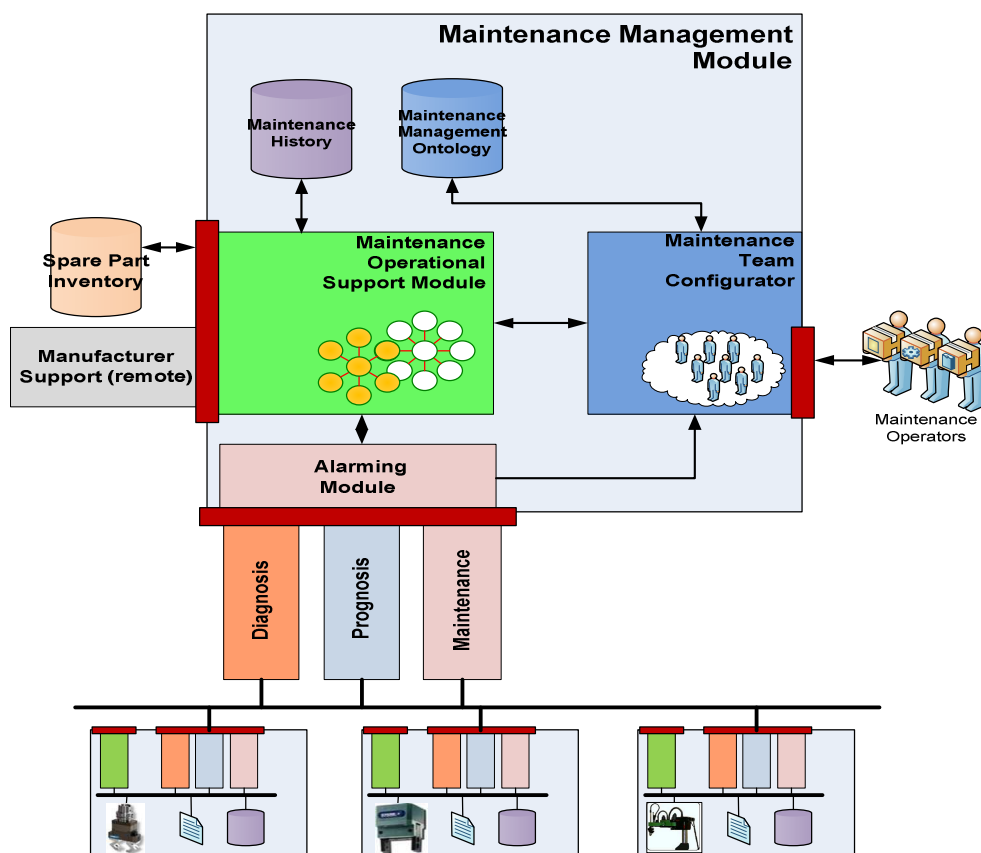


Figura 4.14 - Visão Global da Arquitectura.

A arquitectura apresentada é essencialmente composta por diversos módulos que desempenham papéis de diferente natureza.

O sistema de gestão de manutenção é responsável pelo tratamento de toda a informação de entrada no sistema e disponibiliza interfaces genéricas de comunicação para acesso a dados externos e interacção com o próprio sistema de gestão. Os dados externos incluem a informação de diagnóstico, prognóstico e manutenção reportados pela notificação de eventos, mas pode incluir também informação de *stocks (spare parts inventory)*.

As interfaces genéricas de comunicação disponibilizadas são interfaces para o acesso remoto de operadores situados num centro remoto de manutenção, e de ligação com centros de suporte para consulta ou extracção de informação adicional de fabricantes de equipamentos ou dispositivos.

De seguida, são caracterizados os principais módulos que constituem o sistema de gestão de manutenção:

- ***Alarming Module (AM)*** – Módulo responsável pelo fluxo de entrada de informação para o sistema de manutenção transmitido pelos diversos sistemas registados na plataforma de manutenção. As informações podem ser mensagens de alarme ou de recomendação dependendo do tipo de situação e sistema que originou o pedido. As mensagens recebidas são guardadas e sinalizadas mediante critérios de importância e priorização, até serem consumidas pelo ***Maintenance Team Configurator (MTC)*** e pelo ***Maintenance Operational Support Module (MOSM)***. Uma mensagem com maior carácter de urgência e importância é consumida em primeiro lugar.
- ***Maintenance History Database (MHD)*** – Todo o histórico de eventos e operações reportados são armazenados neste módulo. Cada registo inclui a data e natureza do evento, tal como as acções tomadas e relatórios dos técnicos

(descrevendo as principais decisões se os procedimentos de manutenção recomendados não são adequados). Guardar as evoluções de estado dos sistemas são importantes para a análise dos dados e formulação de recomendações, bem com a definição de procedimentos de manutenção durante o ciclo de gestão.

- ***Maintenance Management Ontology (MMO)*** – Mapeia as principais relações e conceitos que descrevem o sistema, incluindo: a caracterização dos actores de manutenção e suas funções, a definição dos serviços disponibilizados pelo sistema, regras para designação do papel de cada interveniente durante a formação de equipas de manutenção baseados em factores tais como: a experiência, os níveis de confiança, a eficácia e a produtividade de cada interveniente em operações de manutenção, etc.
- ***Maintenance Operational Support Module (MOSM)*** – Este módulo é responsável pelo consumo de mensagens provenientes do **AM** e tratamento de dados num processo de extracção de dados – “*data mining*” (Figura 4.15). Este processo, juntamente com o histórico de operações do equipamento transmitido pelo **MHD** e a correlação da topologia da rede através das falhas reportadas, é possível gerar conhecimento que permite:
 - Implementar um sistema de manutenção preventiva e condicionada;
 - Identificação de correlações de dados e padrões nos eventos reportados;
 - Detectar anomalias e prováveis problemas através de um mapeamento de erros pré-definidos pelo fabricante do produto e/ou experiência do utilizador do sistema;

- Notificar aos elementos da equipa de manutenção, dados técnicos acerca do equipamento em questão.
- Recomendar e renegociar com os técnicos de manutenção as tarefas e procedimentos baseado na sua experiência.
- Oferecer um serviço de suporte operacional on-line, com actualizações de dados e estados de manutenção permitindo uma reavaliação do sistema e redefinição das medidas de manutenção a tomar.

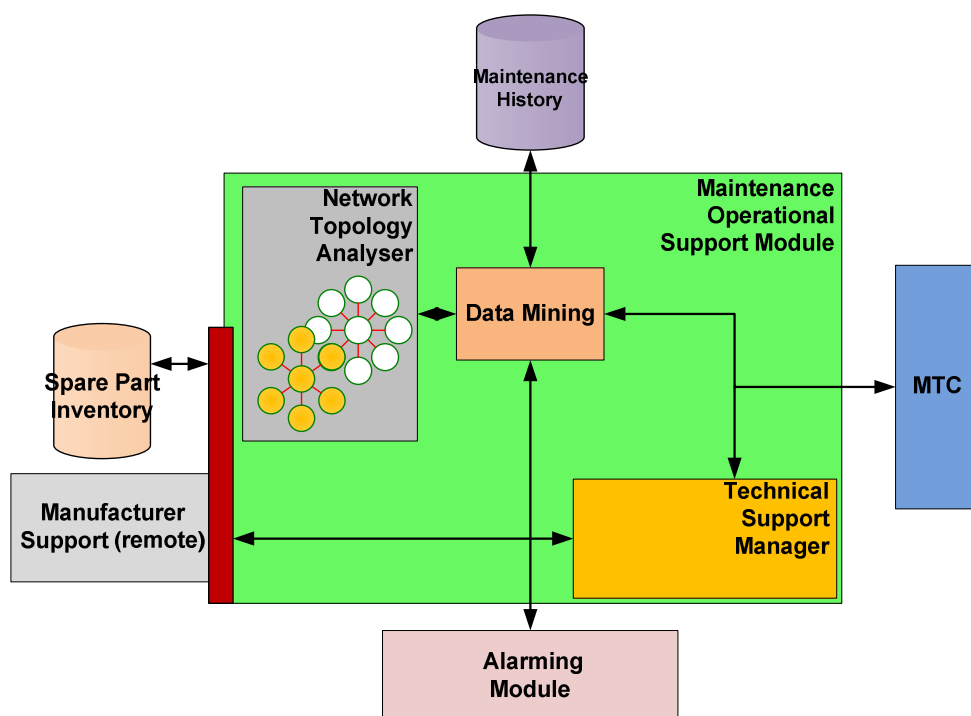


Figura 4.15 - Módulo de Suporte Operacional de Manutenção.

Para além disso, o acesso a dados externos, como o inventário de *stocks*, pode ser acedido através de serviços disponibilizados em interfaces apropriadas.

- **Technical Support Manager (TSM)** – Funcionalidade adicional do MOSM que permite abstrair cada membro da equipa de manutenção como uma peça de *software* e permitir a interacção com o restante sistema, através desta entidade, de acordo com as recomendações do módulo **MTC**.

- **Maintenance Team Configurator (MTC)** - Esse módulo permite a configuração de um grupo de técnicos/operadores e efectuar a gestão de acordo com as suas habilidades, características, tarefas de manutenção/reparação pendentes e experiência (Figura 4.16).

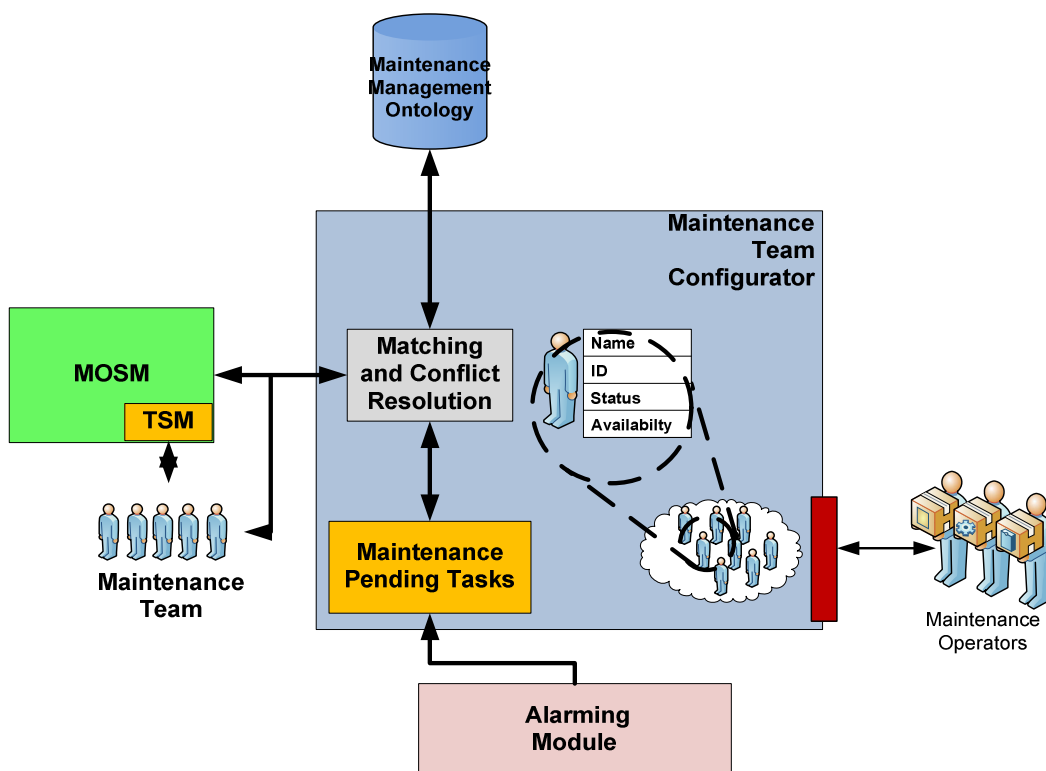


Figura 4.16 - Módulo de Configuração de Equipas de Manutenção

Sempre que uma recomendação ou alarme de reparação é recebida, é destacado um técnico para condução da operação de manutenção e é informado ao **MOSM**.

Por sua vez o **MOSM** sugere uma equipa de manutenção baseada nos intervenientes anteriores e no histórico de manutenções dos dispositivos em falha.

O **MTC** relaciona esta informação de acordo com o conhecimento do **MMO** e da disponibilidade e experiência de cada técnico. Conflitos existentes neste relacionamento são resolvidos pelas regras do **MMO** até a melhor equipa ser constituída. O **MOSM** pode ainda negociar a inclusão de mais ou menos operadores para a tarefa de manutenção.

A formação da equipa de manutenção compreende:

- Avaliar a urgência das tarefas pendentes de manutenção;
- Atribuição e negociação do papel de cada interveniente;
- Notificação aos participantes alocados da natureza e localização da operação de manutenção;

No entanto, o **MTC** pode não conseguir formar uma equipa óptima. De acordo com a urgência da tarefa, um membro menos especializado pode estar presente na equipa. Caso não seja encontrado elementos para formar uma equipa e executar a tarefa de manutenção, o **MTC** pode recomendar ao **TSM** o estabelecimento de uma ligação remota com o centro de manutenção do fabricante, pedindo a alocação de novos recursos ou ajuda personalizada para a execução remota da tarefa.

O módulo pode alternativamente adiar a execução de manutenção com o conhecimento de que um técnico estará disponível num razoável intervalo de tempo.

5 Implementação

Este capítulo apresenta os desenvolvimentos práticos efectuados no sistema inteligente de manutenção, num protótipo que serve de prova de conceito. O capítulo detalha os vários módulos e agentes constituintes do sistema, a integração da plataforma **WSIG** para publicação de serviços dos agentes **JADE** em **Web Services**, a definição de cada serviço, a base de conhecimento de manutenção em documentos **XML**, as interfaces gráficas da plataforma e uma aplicação cliente que permite testar as funcionalidades deste protótipo.

5.1 Funcionalidades dos Agentes

Com base na arquitectura detalhada no capítulo anterior, o sistema de manutenção é composto por vários módulos inteligentes. Nesta primeira abordagem de desenvolvimento, os 3 principais módulos foram implementados e integrados com a plataforma **JADE WSIG** servindo de prova de conceito para a arquitectura descrita. As principais funcionalidades de cada módulo são apresentadas na seguinte Tabela 5.1.

Módulo	Agente	Funcionalidades
Alarming Module	Interface Agent	- Registo e publicação dos serviços de manutenção na plataforma WSIG;

		<ul style="list-style-type: none"> - Consumo das mensagens dos Web Services publicados; - Registo e desregisto dos módulos inteligentes; - Alocação prévia de um técnico de manutenção disponível para cada report de diagnóstico e prognóstico; - Despacho das mensagens para os módulos MTC e MOSM.
Maintenance Team Configurator Module	MTC Agent	<ul style="list-style-type: none"> - Gestão da equipa de manutenção, alocando recursos para as tarefas de manutenção. - A alocação de recursos é feita mediante o número de técnicos solicitados, disponibilidade e os seguintes critérios de selecção: <ul style="list-style-type: none"> ● Histórico de intervenções no respectivo equipamento; ● Avaliação interna na equipa de manutenção;

		<p>- Notificação de alocação prévia ao técnico de manutenção seleccionado com informação do equipamento e tipo de relatório.</p>
Maintenance Operational Support Module	MOSM Agent	<p>- Consome os <i>reports</i> (manutenção, diagnóstico ou prognóstico) do AM, com indicação do técnico aconselhado para a tarefa de manutenção.</p> <p>- Para cada tipo de <i>report</i> é efectuado uma análise dos dados com base na seguinte informação:</p> <ul style="list-style-type: none"> ● Especificação do equipamento, incluindo informação de manutenção; ● Listagem de erros conhecidos <i>a-priori</i>. ● Histórico de intervenções de manutenção no equipamento <p>- Em caso de necessidade, é solicitado ao MTC uma listagem de técnicos disponíveis formando uma equipa de manutenção.</p> <p>- Cada membro da equipa de manutenção é</p>

		<p>notificado com a seguinte informação:</p> <ul style="list-style-type: none"> ● Perfil do técnico de manutenção; ● Informação do <i>report</i>; ● Especificação do equipamento, se existente; ● Identificação do problema, se reconhecido. ● Listagem de operações a desempenhar, se identificado o problema.
--	--	--

Tabela 5.1 - Descrição das Funcionalidades dos Agentes

A utilização da tecnologia agentes adequa-se de acordo com as características previamente definidas para esta arquitectura, adquirindo assim capacidades de autonomia, raciocínio, comunicação e interoperabilidade importantes para a gestão de recursos e produtos e suporte operacional de tarefas de manutenção.

Para além dos agentes descritos, foram definidos diversos serviços (Tabela 5.2) que permitem a interacção com o sistema de gestão de manutenção entre os seus utilizadores (equipamentos, máquinas, robôs, etc.), e a recepção de notificações de alarmes de cada sistema registado.

DEVICE_REGISTRATION	Serviço de notificação de registo de um equipamento no sistema de manutenção.
---------------------	---

DEVICE_UNREGISTRATION	Serviço de notificação de desregisto do equipamento no sistema de manutenção.
DIAGNOSIS_REPORT	Serviço de notificação de <i>report</i> de diagnóstico.
PROGNOSIS_REPORT	Serviço de notificação de <i>report</i> de prognóstico.
MAINTENANCE_REPORT	Serviço de notificação de <i>report</i> de manutenção.
HEART_BEAT	Serviço de sinalização de controlo de um equipamento no sistema de manutenção.

Tabela 5.2 - Serviços de comunicação do sistema inteligente de manutenção

Cada um destes serviços é integrado numa interface genérica de comunicação através da publicação de um **Web Service**. Este **Web Service** é resultado da integração da tecnologia dos agentes com a publicação de serviços na plataforma **WSIG**, aumentando assim a versatilidade e interoperabilidade nas comunicações com o sistema de manutenção e a plataforma de multi-agentes, independentemente da tecnologia que o cliente usa. Esta implementação será detalhada mais em pormenor de seguida.

5.2 Integração dos agentes JADE com Web Services

No módulo **AM**, foi definido um agente serviço (*Interface Agent*) responsável pela publicação de serviços de manutenção no directório de serviços de ‘páginas amarelas’ da plataforma **JADE**. Mas com a integração da plataforma *JADE Web Service Integration Gateway* (**WSIG**) e a plataforma de agentes **JADE** é possível traduzir esses mesmos serviços em **Web Services**.

A vantagem inerente da utilização de **Web Services** nesta arquitectura é facilitar, nomeadamente a comunicação entre aplicações em plataformas diferentes, isto é, em plataformas não baseada em agentes ou tecnologia Java.

No entanto, para poder integrar os **Web Services** com os serviços de agentes **JADE** foi necessário definir programaticamente uma ontologia que mapeia a definição de cada serviço individualmente e registá-la no mesmo momento em que é publicado o serviço de manutenção no directório de serviços (Figura 5.17).

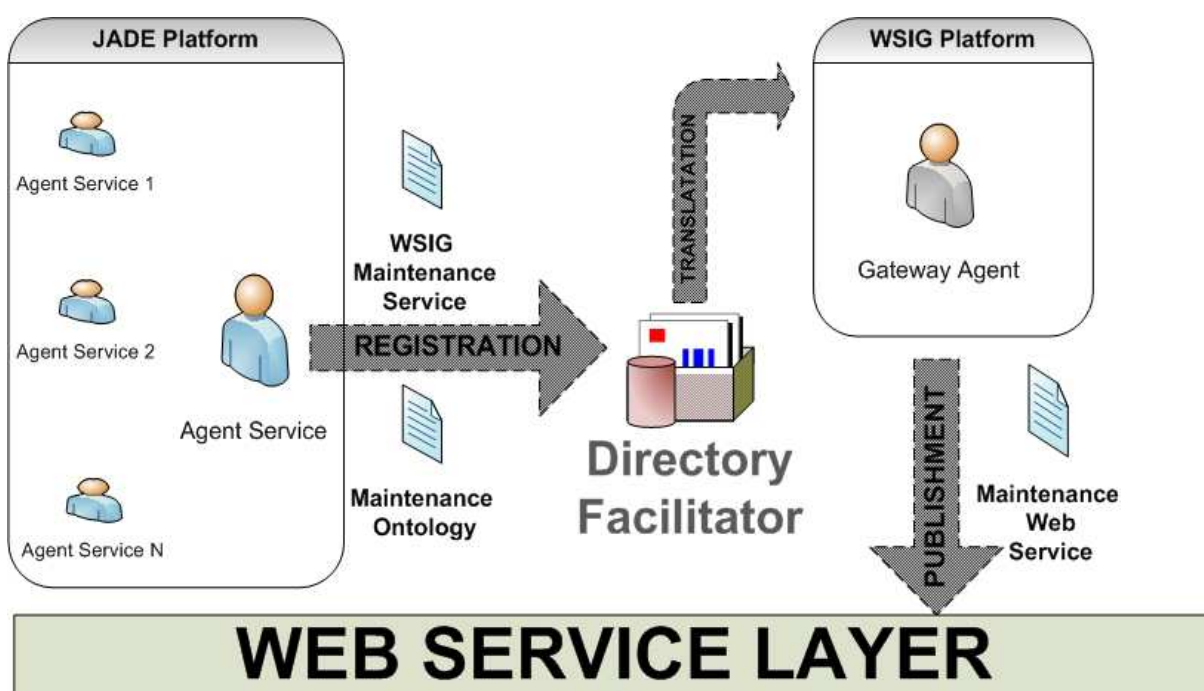


Figura 5.17 - Publicação de um serviço de manutenção como Web Service

Outra vantagem importante para a utilização de uma ontologia é possibilitar a comunicação de informação complexa entre as plataformas de agentes, delineando uma sintaxe perceptível para cada receptor da informação, de acordo com as especificações da **FIPA**.

Como a plataforma de agentes é baseada numa implementação Java, a estrutura de informação complexa de cada serviço de manutenção é representada por intermédio de objectos **Java** (Figura 5.18).

```

public class ReportingDiagnosis implements AgentAction, Serializable{

    private String uniqueId;
    private Integer messageType;
    private String description;
    private Integer severity;
    private Integer uptime
    private String timeStamp;
    private List faultsHistory;
    private List currentFaults;
    private String diagnosticCode;
    private String diagnosticDescription;

    public String getUniqueId() { return uniqueId; }
    public void setUniqueId(String uniqueId) { this.uniqueId = uniqueId; }
    public Integer getMessageType() { return messageType; }
    public void setMessageType(Integer messageType) {
        this.messageType = messageType; }

    ....

}

```

Figura 5.18 – Serviço de notificação do evento Diagnóstico (Classe JAVA)

Portanto, a representação de cada serviço na especificação do **Web Service** na ontologia de manutenção é bastante simples e clara (Figura 5.19).

```

public class MessageOntology extends Ontology implements MessageVocabulary {

    ....

    public MessageOntology() {
        super(ONTOLOGY_NAME, BasicOntology.getInstance());

        try {

            add(new AgentActionSchema(DIAGNOSIS_REPORT),

```

```

ReportingDiagnosis.class);
....
}
}
....
}

```

Figura 5.19 – Especificação do serviço de notificação de diagnóstico na Ontologia de Manutenção

5.3 Base de conhecimento de suporte à Manutenção

De modo a suportar as funcionalidades dos agentes, foi criado um conjunto de ficheiros **XSD** que definem a estrutura dos elementos e atributos para documentos XML. Estes documentos **XML** são utilizados para armazenar informação como base de conhecimento de suporte à manutenção.

No seguimento do trabalho desenvolvido, foram definidos os seguintes *schemas*:

- **Schema** (*deviceRegistrationBD.xsd*) para o documento **XML** (*deviceRegistrationBD.xml*) de armazenamento de registos de equipamentos e dispositivos no sistema de manutenção. Cada registo inclui uma descrição geral do equipamento ou dispositivo (nome, tipo de equipamento, número de série, etc.), um histórico de falhas ou problemas previamente identificados, e uma lista de sistemas de acoplados ao dispositivo registado. Na subscrição de cada dispositivo na plataforma é gerado um identificador chave único de registo necessário para o envio de eventos de sinalização e de *reports*.

- **Schema** (*maintenanceTeamBD.xsd*) para o documento **XML** (*maintenanceTeamBD.xml*) com um repositório de informação acerca de técnicos de manutenção registados na plataforma de manutenção. Para cada técnico é guardado uma descrição geral, uma lista de competências e um histórico de intervenção. Do mesmo modo, para cada técnico é gerado um identificador chave único.

- **Schema** (*deviceSpecificationBD.xsd*) para o documento **XML** (*deviceSpecificationBD.xml*) de descrição da especificação de equipamentos. A documentação técnica e especificação funcional de cada dispositivo podem ser registadas na plataforma de manutenção. Esta informação é importante e crucial, pois permite ao sistema de manutenção avaliar dados de manutenção de cada equipamento registado no sistema de manutenção e correlacionar dados de *reports* de diagnóstico, prognóstico ou até mesmo *reports* de manutenção com esta base de conhecimento. Para além da sua importância na monitorização de operações de manutenção pelo sistema, estes dados podem ser consultados remotamente por todos os técnicos em situações de suporte operacional.

- **Schema** (*errorMapping.xsd*) para o documento **XML** (*errorMapping.xml*) com o mapeamento de erros e problemas *a priori* conhecidos, e uma listagem de principais soluções. Esta base de conhecimento pretende facilitar e agilizar a detecção de eventuais problemas já detectados pela experiência de cada técnico de manutenção. Estes dados estão divididos em três categorias (diagnóstico, prognóstico ou manutenção) podendo ser utilizados na análise de cada tipo de *report*. No entanto, cada tipo de erro ou problema é identificado pela categoria do erro e respectivo código de erro. Esta base de conhecimento pode ser igualmente consultada remotamente por qualquer operador de manutenção.

A definição de cada ficheiro XSD pode ser consultada no anexo I.

Para além das vantagens e desvantagens dos documentos **XML** mencionadas anteriormente, a sua utilização foi direccionada para facilitar o armazenamento de dados de manutenção e permitir uma rápida integração com a tecnologia dos agentes **JADE**.

Neste contexto, os documentos **XML** foram a solução mais adequada e flexível para lidar com a manipulação de dados, e os ficheiros **XSD** deram uma maior definição e clareza na estruturação dos dados.

Outro aspecto relevante foi a consulta de dados poder ser facilmente implementado baseando-se apenas na interpretação dos ficheiros **XML** e disponibilizá-los para o utilizador, ao invés de utilizar tecnologias de base de dados relacionais que forçaria um maior esforço em determinar e implementar métodos de consultas para uma **BD**.

É certo que, para trabalhar com grandes volumes de dados as tecnologias de **BD** dão um melhor desempenho do que interpretar grandes quantidades de dados em ficheiros, e a estruturação de dados não é tão rígida como definida para os documentos **XML**.

No caso de se verificar um aumento de complexidade dos sistemas ou manipulação dos dados, os documentos **XML** podem ser facilmente integrados e armazenados em **BDs**.

5.4 Interfaces Gráficas

A informação da base de conhecimento de suporte à Manutenção é disponibilizada para consulta através da implementação de várias interfaces gráficas. Como apresentado anteriormente, a base de conhecimento é principalmente constituída por um conjunto de 4 ficheiros **XML**.

Desse modo, a interface principal (Figura 5.20) permite ao utilizador optar qual o conjunto de informação a visualizar. Individualmente, cada interface é mostrada e detalhada de seguida.



Figura 5.20 - Interface Gráfica principal de Consulta à Base de Conhecimento de Suporte à Manutenção

5.4.1 Consulta de Operadores de Manutenção

A 1ª opção da interface gráfica principal permite aceder à informação de operadores registados na plataforma de Manutenção.

A partir da interface de consulta de operadores, (Figura 5.21) podemos visualizar a lista de operadores de manutenção registadas no sistema, bem como a disponibilidade de cada técnico.

Esta interface é útil, uma vez que permite mostrar a informação utilizada pelo módulo **MTC** responsável pela gestão de alocações de operadores para intervenções de Manutenção.



Figura 5.21 - Lista de Operadores registados na plataforma de Manutenção

Para além da lista de operadores, é possível visualizar informação acerca das competências (Figura 5.22) e do histórico de operações de Manutenção (Figura 5.23) através da selecção de um dos operadores.

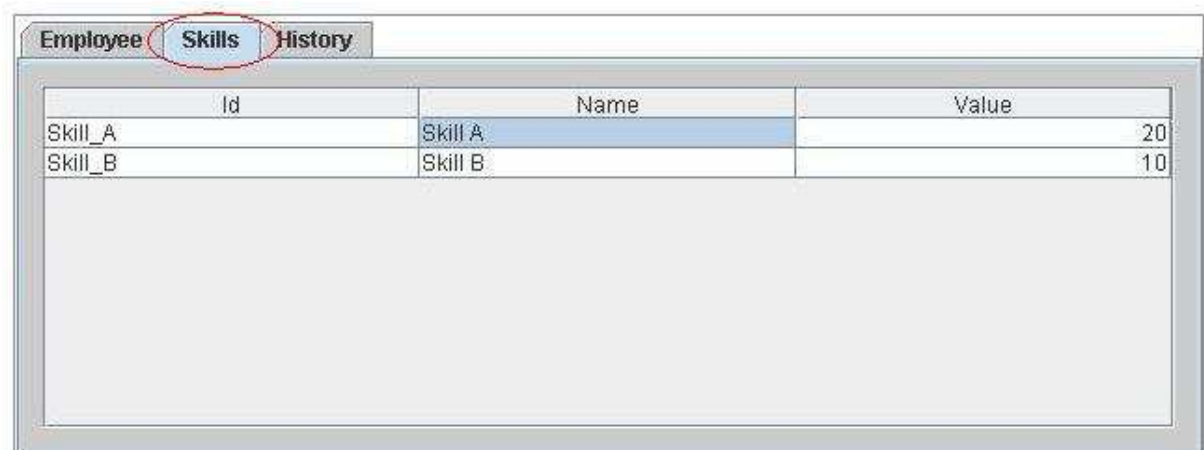


Figura 5.22 - Lista de competências de um operador de Manutenção

Employee Skills History		
Id	Name	Type
1	Operation A	Functional
2	Operation B	Technical
3	Operation C	Maintenance Revision

Figura 5.23 - Histórico de Operações de Manutenção de um operador

Adicionalmente, a informação apresentada nesta interface pode ser consultada mais em detalhe através de interfaces particulares, como por exemplo os detalhes de uma intervenção operada por um técnico registado na plataforma (Figura 5.24).

MAINTENANCE OPERATION HISTORY v1.0

OPERATION HISTORY

Operation ID: 1
Device ID: 4

Name: Operation A

Type: Functional

Status: Started

Start Date: 2008-11-09T00:00:00

Pending Date: 1900-01-01T00:00:00

End Date: 1900-01-01T00:00:00

Cost: 100.0

Reason:

Note:

Figura 5.24 - Exemplo dos detalhes de uma operação de Manutenção

5.4.2 Consulta de Especificações de Equipamentos

A base de conhecimento com a especificação documental de equipamentos é das mais importantes para o funcionamento deste sistema.

Tanto utilizado para dar suporte e informação de funcionamento de cada equipamento nas operações de Manutenção, como para avaliar e cruzar dados de eventos reportados, o sistema fornece uma interface de consulta de especificações (Figura 5.25).



DEVICE SPECIFICATION INFORMATION v1.0

FCT FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

DEVICE SPECIFICATION INFORMATION

Device Type: robots

Description Global Properties Maintenance Properties

Name: AX-R15

Model: ABB Family: AX

Serial Number: 23456788765 Port Number: 876565AS65

Manufacturer Details:

Manufacturer: ABB Construction Date: 2008-11-01T00:00:00

Address: Lisbon

Contact: 917794670

Figura 5.25 - Interface Gráfica para Consulta de Especificações de Equipamentos

A interface permite filtrar a informação sobre especificações por tipos de equipamentos. Diversas máquinas, dispositivos, ou equipamentos têm especificações semelhantes apresentando características e propriedades iguais entre elas.

Na janela inicial, a informação é detalhada com dados de fabricante e descrições gerais dos equipamentos em questão. Para além da descrição geral de cada dispositivo, podemos visualizar várias propriedades relevantes para o funcionamento adequado e boa rentabilidade de cada equipamento.

As características visualizáveis são propriedades eléctricas, de desempenho, pneumáticas e físicas (Figura 5.26).

Outras propriedades poderiam ser adicionadas, mas apenas as propriedades mais relevantes foram projectadas para este protótipo.

Global Properties				
Electrical	Performance	Environment	Pneumatic	Physical
Posistion Repeatibility: 5				
Response Time: 258				

Figura 5.26 - Janela de Propriedades Globais.

Por fim, uma janela de propriedades de manutenção está disponível para visualização (Figura 5.27). Esta informação é importante para tratar e supervisionar a manutenção de cada dispositivo registado na plataforma de Manutenção e operado pelo módulo **MOSM**.

Property	Value	Unit
Mean Time To Failure (MTTF):	10	(days)
Mean Time Between Failure (MTBF):	20	(days)
Mean Time To Repair (MTTR):	30	(days)
Uptime:	40	(minutes)
Idle Time:	100	(minutes)
Unhandled Events:	5	(events)

Figura 5.27 - Janela de Propriedades de Manutenção

5.4.3 Consulta de Mapeamento de Erros e Problemas

De seguida outra interface gráfica é disponibilizada no sistema de Manutenção e refere-se à consulta de mapeamento de erros e problemas (Figura 5.28).

Cada mapeamento descreve e identifica um erro ou problema *a priori* conhecido pelo sistema de Manutenção. Este mapeamento é importante, uma vez que permite identificar que situação de erro é reportada nos eventos de diagnóstico, de prognóstico ou de manutenção recebidos dos dispositivos registados na plataforma de Manutenção.

Os tipos de erros ou problemas mapeados são categorizados em três grupos (diagnóstico, prognóstico, e manutenção). Esta base de conhecimento de erros ou problemas é resultante do conhecimento e experiência de cada operador construindo assim uma base de dados que, permite detectar e acelerar o reconhecimento de erros pelo sistema de manutenção.

Para além da descrição de cada erro ou problema reconhecido pelo sistema, uma lista de soluções ou recomendações é anexa e permite dar suporte aos técnicos e operadores durante as intervenções de Manutenção.

ERROR MAPPING INFORMATION v1.0

FCT FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

ERROR MAPPING INFORMATION

Error Type: Maintenance Error Code: 305 - Substituição das válvulas

Solutions

Nº.	Maintenance Solution
1	Agendamento com o cliente data de manutenção
2	Garantir os recursos e equipamentos para operação de sistema
3	Proceder à alocação da equipa de intervenção na manutenção do sistema
4	Proceder à substituição das válvulas
5	Proceder a testes elementares
6	Testes de qualidade e segurança
7	Preenchimento do formulário de manutenção do sistema
8	Restabelecimento à normalidade do sistema

More Info

Figura 5.28 - Interface Gráfica para consulta de Mapeamentos de Erros

Cada solução ou recomendação pode ser consultada em detalhe através de uma interface individual (Figura 5.29).

The screenshot shows a software window titled "ERROR DETAIL v1.0". Inside, there is a section titled "ERRORHANDLING DETAIL" in a yellow box. Below this, there are three input fields: "Error Type:" with the value "Maintenance", "Error Code:" with the value "305", and "Error Description:" with the value "Substituição das válvulas". At the bottom, there is a "Solution Description" section containing the text "Garantir os recursos e equipamentos para operação de sistema".

Figura 5.29 - Consulta detalhada de uma solução de intervenção para um problema mapeado

5.4.4 Consulta de Equipamentos registrados no Sistema de Manutenção.

Por último, esta interface permite ao operador de Manutenção consultar os detalhes de subscrição de um equipamento com a informação geral, o histórico de falhas e a lista de sistemas acoplados ao dispositivo através do identificador único de registo (Figura 5.30).

A lista de sistemas acoplados dá informação adicional acerca do tipo de equipamento registado. Neste contexto, um sistema acoplado pode ser definido como uma parte agregada de um sistema mais complexo constituído por diversos sub-sistemas ou vários equipamentos.

O histórico de falhas permite aos operadores de Manutenção obterem *feedback* acerca do historial de problemas identificados no equipamento registado nesta plataforma de Manutenção. Um longo historial de falhas pode indicar ao operador que o equipamento necessita de uma atenção mais frequente, agendando tarefas de supervisão e manutenção com

uma periodicidade mais curta. Cada falha pode ser consultada com maior detalhe através de uma interface de individual de consulta (Figura 5.31).

DEVICE REGISTRATION INFORMATION v1.0

FCT FACULDADE DE CIÊNCIAS E TECNOLOGIA UNIVERSIDADE NOVA DE LISBOA

Device Registration Information

Registration ID: 3

Name: ABB-XRF Serial Number: 123

Friendly Name: ABB Port Number: 1232545

Device Type: 3124245

Description:

Associated Systems:

System Id:
23984

Faults History:

Fault Id:	Fault Info:	Fault Type
123874	Falha Mecânica	TECHNICAL

Figura 5.30 - Interface Gráfica para Consulta de Equipamentos no Sistema de Manutenção

Fault Description GUI v1.0

FCT FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

Fault Description

Fault Id: 123874

Fault Description:
Falha Mecânica

Fault Type: TECHNICAL

TimeStamp(yyy-MM-dd HH:mm:ss): 2008-12-12 11:59:00

Device Description

Source Id: 23455

Serial Number:

Figura 5.31 - Detalhes de uma falha ocorrida no equipamento

5.5 Caso de Teste

Para provar as funcionalidades do sistema inteligente de manutenção e testar a integração de **Web Services** com os agentes **JADE**, de seguida é apresentado uma aplicação cliente (utilizador do serviço de Manutenção) com interfaces gráficas que simula um determinado dispositivo ou equipamento na invocação do **Web Service** e na interacção com o Sistema de Manutenção.

Mas antes de detalhar e aprofundar a aplicação cliente, é primeiramente necessário indicar as configurações para inicialização da plataforma **WSIG**.

5.5.1 Configuração da plataforma **WSIG**

Para inicializar a plataforma **WSIG**, uma vez que se trata de uma aplicação que usa tecnologia de agentes, é necessário arrancar com a plataforma de agentes **JADE**.

Após isso, a solução **WSIG** juntamente com a definição das classes e ontologias de Manutenção e ficheiro de configuração **WSIG**, os quais são empacotadas numa aplicação Web (pacote `wsig.war`) através da execução de um ficheiro de construção **ANT Buildfile** e instalado num servidor aplicacional **JBoss**.

O ficheiro de configuração **WSIG** é imprescindível para o funcionamento da plataforma **WSIG** e contém as principais propriedades (Tabela 5.3).

Propriedade	Descrição	Valor
Host	Endereço do container da plataforma JADE	93.108.9.185
Port	Porto do container da plataforma JADE	1099

container-name	Nome do container da plataforma WSIG	WSIG-Container
local-port	Porto local do container da plataforma WSIG	1200
wsig.agent	Classe de implementação Agente Gateway WSIG	com.tilab.wsig.agent.WSIGAgent
wsig.uri	Endereço URL para invocar a plataforma WSIG.	http://93.108.9.185:8080/wsig/ws
wsig.console.uri	Endereço URL para invocar a consola WSIG.	http://93.108.9.185:8080/wsig
wsig.timeout	Tempo limite (em milisegundos) para execução das acções dos agentes correspondentes às invocações WSIG	30000
wsdl.localNamespacePrefix	Prefixo TNS usado na construção do WSDL	Impl
onto.messageOntology	Classe de descrição da Ontologia WSIG	pt.unl.fct.maintenance.wsig.ontologies. MessageOntology

Tabela 5.3 - Propriedades principais de configuração da plataforma WSIG

Findo a instalação da plataforma **WSIG** no servidor aplicacional, a consola **WSIG** está disponível para consulta das propriedades de configuração **WSIG** e a lista de **Web Services** publicados pelos agentes **JADE** (Figura 5.32).

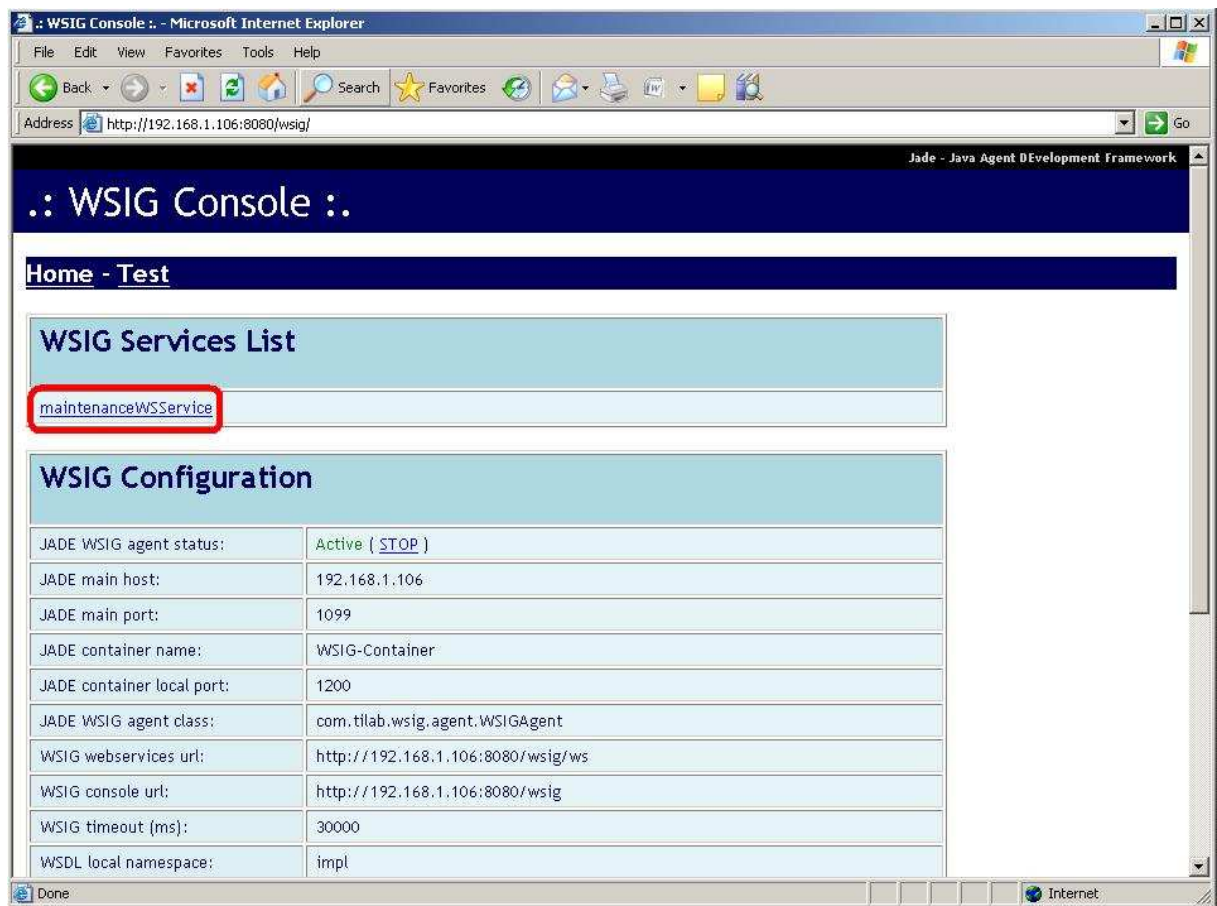


Figura 5.32 - Consola Web da plataforma WSIG

Como mencionado anteriormente, o módulo **AM** contém o agente responsável pela publicação de Manutenção, que é convertido em **Web Service**.

A particularidade em relação à publicação normal de serviços no DF é a adição da propriedade **wsig** (Figura 5.33) que indica ao **DF** a exposição do agente serviço como **Web Service**, e da propriedade **wsig-mapper** para registo da ontologia de Manutenção na plataforma **JADE**.

```
sd.addProperties(new Property("wsig", true));
sd.addProperties(new Property("wsig-
mapper",agentVocabulary.MESSAGE_ONTOLOGY_ROOT_PATH));
```

Figura 5.33 - Propriedade para exposição de agentes serviço em Web Services

5.5.2 Aplicação Cliente do Sistema Inteligente de Manutenção

Após a implementação da arquitectura proposta no capítulo 4 como prova de conceito, foi criada uma aplicação cliente de interacção com o Sistema Inteligente de Manutenção.

A aplicação cliente permite invocar cada serviço exposto no Sistema de Manutenção como Web Service, simulando o comportamento de um determinado equipamento ou dispositivo.

5.5.2.1 Serviço de Registo de Equipamento

Qualquer dispositivo para interagir com o Sistema de Manutenção deve proceder ao registo associando-se ao sistema (Figura 5.34).

The screenshot displays the 'Maintenance Web Services Client GUI v1.0' window. The title bar includes 'File' and 'About' menus. The main content area features the logo of the 'FACULDADE DE CIÊNCIAS E TECNOLOGIA UNIVERSIDADE NOVA DE LISBOA' and a green 'REGISTER' button. Below this, there is a 'Device Registration' section with input fields for 'Name', 'Serial Number', 'Friendly Name', 'Port Number', 'Description', and 'Device Type'. At the bottom, there are two panels: 'Associated Systems' with a 'System Id' field and a list box, and 'Faults History' with a table header showing 'Fault Id', 'Fault Info', and 'Fault Type'.

Fault Id	Fault Info	Fault Type
----------	------------	------------

Figura 5.34 - Interface Cliente para Registo de Equipamentos no Sistema de Manutenção

Cada campo da interface corresponde aos parâmetros do serviço *deviceregistration* publicado. Os dados de entrada correspondem à descrição geral do equipamento, à lista de sistema acoplados e ao histórico de falhas.

No registo, cada equipamento é identificado por uma chave única que é devolvida na invocação do serviço atrás mencionado. A partir daí, qualquer interacção com o sistema é efectuada por intermédio desse mesmo identificador (Figura 5.35).



Figura 5.35 - Registo do Equipamento com sucesso

O serviço de registo operado pelo agente serviço é retratado pelo seguinte fluxograma (Figura 5.36).

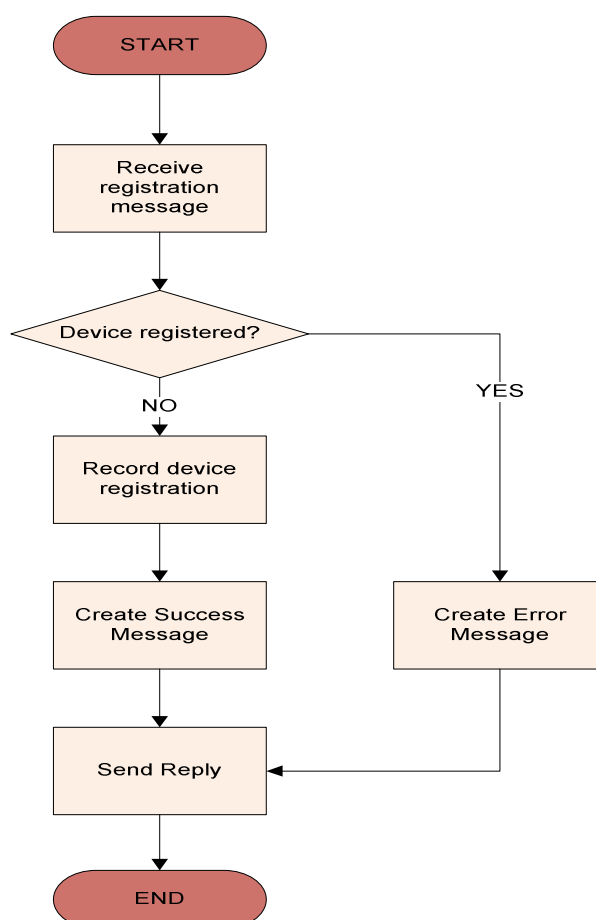


Figura 5.36 - Fluxograma de registo de equipamento no Sistema de Gestão de Manutenção

Após o registo do equipamento e abrindo o menu de *reports* (Figura 5.37) com a informação do *id* do equipamento, podemos simular o envio dos três tipos de *reports* para o Sistema de Gestão de Manutenção.



Reporting Messages GUI v1.0

FCT FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

Reporting Messages

INFO SYSTEM

SYSTEM ID: 5

Type Of Message:

☒ Maintenance Reports

☐ Prognosis Reports

☐ Diagnosis Reports

OK

Figura 5.37 - Interface de simulação de *reports*

5.5.2.2 Serviço de Notificação de *Report* de Diagnóstico

A identificação de problemas em cada equipamento, resultante da avaliação e do diagnóstico de sintomas de anomalias detectadas, é notificada através deste serviço.

Uma interface personalizada (Figura 5.38) permite preencher os dados de entrada da notificação de *report* de diagnóstico, cujo nome do serviço é identificado no Sistema de Gestão de Manutenção por *diagnosisreport*.

Diagnosis Reporting GUI v1.0

FCT FACULDADE DE CIÊNCIAS E TECNOLOGIA UNIVERSIDADE NOVA DE LISBOA

Diagnosis Reporting Send

Diagnosis Report:

System... 5 Message Type: Diagnostic Uptime (ms):

Description: Severity: None

Faults History

Fault Id:	Fault Info:	Fault Type:
12342	XPTO	FUNCTIONAL

Current Faults:

Fault Id:	Fault Info:	Fault Type:
-----------	-------------	-------------

Diagnosis Handler

Diagnosis Code:

Diagnosis Description:

Figura 5.38 - Interface para envio de *reports* de diagnóstico

Na ocorrência de um problema, um *report* de diagnóstico é gerado contendo essencialmente uma listagem do diagnóstico de falhas detectadas e uma avaliação do provável tipo de erro. Através da base de conhecimento de mapeamento de erros, mais informações podem ser transmitidas aos operadores de Manutenção, incluindo detalhes de soluções.

Na recepção da mensagem no Sistema de Gestão de Manutenção, o agente serviço (*Interface Agent*) identifica o tipo de *report* e coloca numa fila de mensagens. Nesta implementação foi colocado uma estratégia de tratamento de mensagens **FIFO**.

Quando retirada da fila o agente serviço solicita a alocação prévia ao agente **MTC** de um operador de manutenção disponível. Seguidamente é transmitido ao módulo **MOSM** o *report* onde será analisado e recolhida informação para notificar cada operador alocado.

Na avaliação de cada *report* de diagnóstico, e mediante o tipo de problema encontrado o agente MOSM procederá à alocação de operadores de manutenção adicionais em situações de intervenções complexas. Após a informação recolhida, cada operador é notificado pelo envio de uma mensagem electrónica (Figura 5.39) de alocação para uma tarefa de Manutenção, identificando o equipamento sujeito à tarefa, detalhes do *report* e do erro detectado, bem como a listagem de recomendações de actividades a serem executadas durante a operação de manutenção.

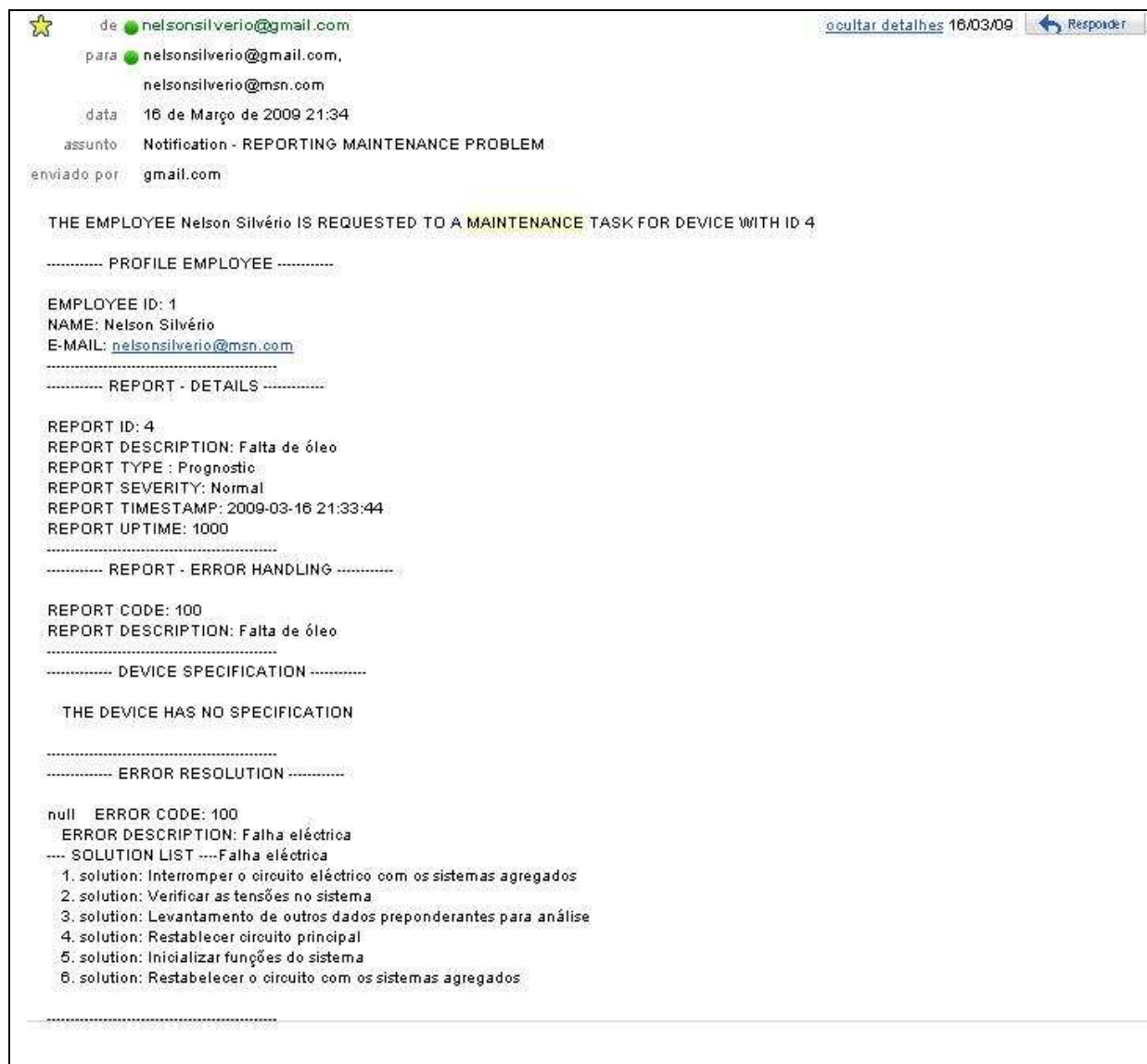


Figura 5.39 - Excerto de e-mail enviado ao operador de manutenção

Na Figura 5.40 é ilustrada um diagrama de sequência UML com a troca de mensagens entre os vários intervenientes numa notificação de *report* de diagnóstico.

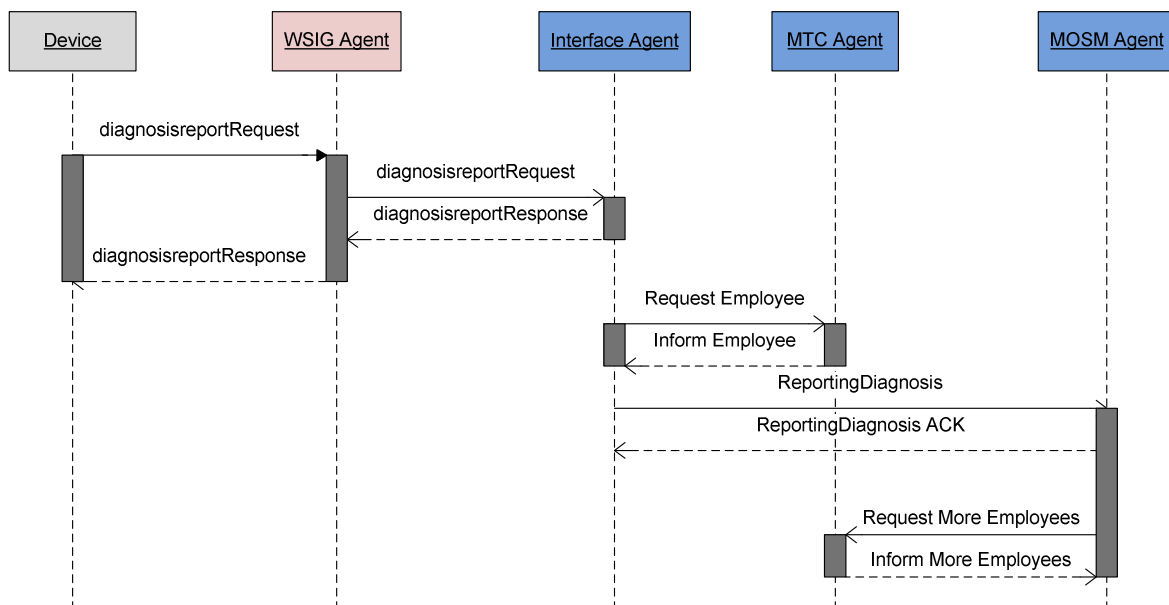


Figura 5.40 - Diagrama de sequência UML da notificação de report de diagnóstico

5.5.2.3 Serviço de Notificação de *Report* de Prognóstico

Este serviço é idêntico ao serviço de diagnóstico, mas difere no tipo de report, nos dados de entrada no serviço, na análise do *report* no módulo **MOSM** e nas mensagens envolvidas durante o processo de notificação.

A interface personalizada é ilustrada na Figura 5.41, onde são colocados os dados de entrada do serviço de *prognosisreport*. Os dados são principalmente compostos pelo prognóstico de um erro conhecido e identificado pela base de conhecimento de erros no Sistema de Gestão de Manutenção.

Figura 5.41 - Interface para envio de *reports* de prognóstico

Na Figura 5.42 é mostrada a sequência de mensagens trocadas entre cada interveniente no processo de notificação de *report* de prognóstico num diagrama de sequência UML.

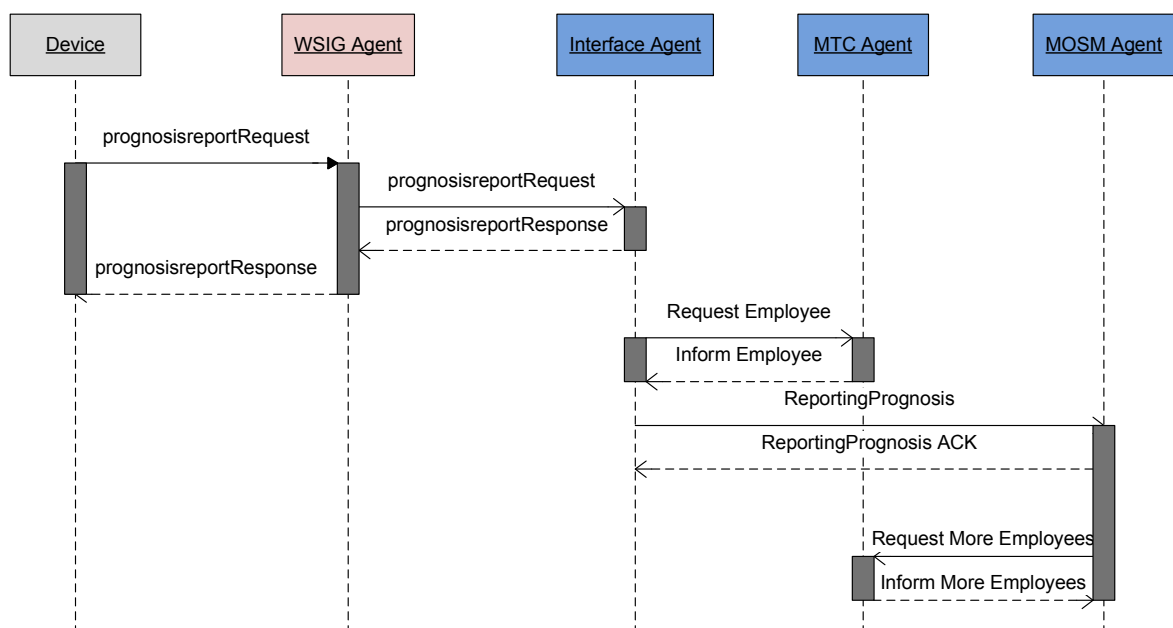


Figura 5.42 - Diagrama de sequência UML da notificação de *report* de diagnóstico

5.5.2.4 Serviço de Notificação de *Report* de Maintenance

O serviço de Manutenção permite informar e alertar o Sistema de Gestão de Manutenção, a necessidade de operar tarefas de manutenção. Este tipo de notificação não tem tanta prioridade como os dois serviços de notificações mencionados atrás, não requerendo assim a necessidade de alocar previamente recursos.

Para simular este tipo de serviço a interface personalizada ilustrada na Figura 5.43 permite introduzir parâmetros de entrada para a chamada do respectivo serviço *maintenancereport*.

The screenshot shows a web application window titled "Maintenance Reporting GUI v1.0". The header features the logo of the "FACULDADE DE CIÊNCIAS E TECNOLOGIA UNIVERSIDADE NOVA DE LISBOA" (FCT). Below the header, the main content area is titled "Maintenance Reporting" in green text. A "Send" button is located in the top right corner. The interface is divided into two main sections: "Diagnosis Report" and "Maintenance Handler".

Diagnosis Report:

- System Id:** A text input field containing the value "5".
- Message Type:** A dropdown menu with "Maint.." selected.
- Uptime (ms):** A text input field.
- Description:** A large text area for entering details.
- Severity:** A dropdown menu with "None" selected.

Maintenance Handler:

- Maintenance Code:** A text input field.
- Maintenance Description:** A large text area for entering details.

Figura 5.43 - Interface para envio de *reports* de manutenção

À semelhança dos outros serviços, um dos parâmetros mais importante deste serviço é a indicação de um código de erro mapeado na Base de Conhecimento introduzido para notificar o tipo de operação que é solicitado para a intervenção de manutenção.

Na Figura 5.44 é mostrada a sequência de mensagens trocadas entre cada agente interveniente no processo de notificação de *report* de manutenção num diagrama de sequência UML.

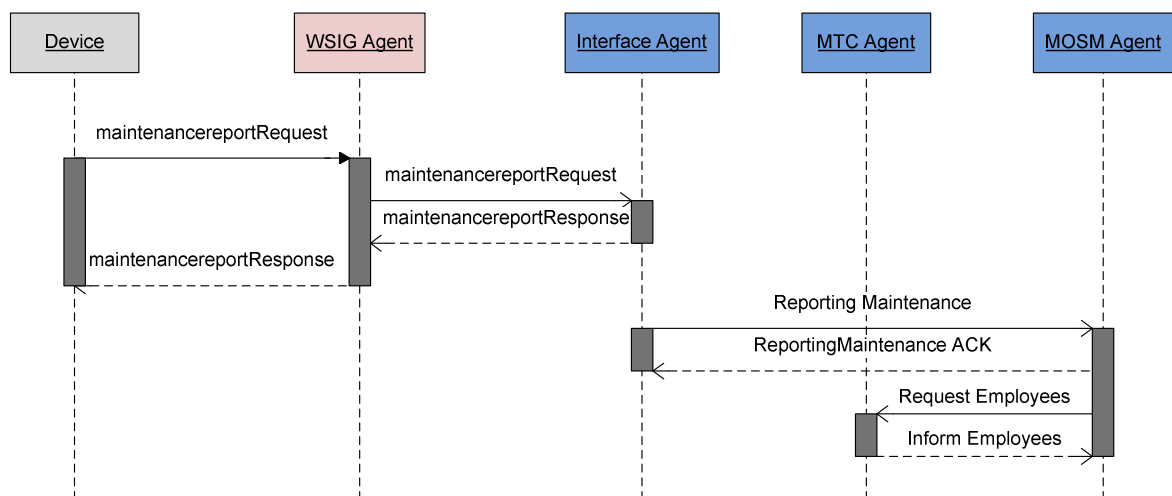


Figura 5.44 - Diagrama de sequência UML da notificação de report de manutenção

5.5.2.5 Serviço de Desregisto de Equipamento

Por fim, o serviço de desregisto desassocia um determinado equipamento do Sistema de Manutenção, através do envio do identificador chave de um equipamento registado. Neste sentido, cada equipamento registado é removido da Base de Conhecimento, impedindo qualquer tipo de interacção com a plataforma de Manutenção.

6 Conclusões e Perspectivas

6.1 Conclusões

A literatura da Manutenção destaca predominantemente a importância em melhorar os aspectos de eficiência, produtividade e de sustentabilidade em todo o ciclo de vida de concepção de um produto realçando assim o papel fundamental que a manutenção desempenha nestas áreas.

Contudo, as questões organizacionais como a gestão de recursos, a gestão de equipas de intervenção e de manutenção e suporte operacional não têm merecido destaque na literatura, sendo evidente a falta de aplicações e de sistemas que permitam melhorar a produtividade e eficiência durante as acções e tarefas de manutenção.

Neste sentido, esta dissertação permitiu a apresentação de uma arquitectura de um sistema inteligente baseada na tecnologia de agentes, que oferece um conjunto de funcionalidades para a gestão de equipamentos e recursos (técnicos, operadores, gestão de stocks. etc.), e suporte operacional de manutenção contribuindo para melhoria da rentabilidade dos processos e tarefas de manutenção.

Do ponto de vista funcional, este sistema foi implementado em parte como uma prova de conceito da arquitectura desenhada focando sobretudo a exploração de características do

conceito de e-Maintenance, a integração de sistemas inteligentes na plataforma e a disponibilização de agentes serviços convertidos em Web Services proporcionando o aumento da interoperabilidade e interligação (mecanismos *Plug and Play*) do sistema independentemente da tecnologia usada pelo utilizador final.

Deste modo, o papel de e-Maintenance nesta arquitectura é considerado como uma manutenção de suporte que permitiu associar as tecnologias de Web Services com a tecnologia de agentes, oferecendo características inteligentes ao sistema através de mecanismos de raciocínio, interacção e colaboração, podendo ser aplicado no contexto dos sistemas de automação industriais.

Assim, o resultado efectivo deste sistema inteligente permitiu contribuir nos seguintes aspectos:

- ✓ Melhorar a gestão de equipamentos e recursos permitindo a detecção de sintomas de anomalias e a prevenção de ocorrência de falhas com base num sistema de alarmística.
- ✓ Dar o suporte operacional viabilizando o aumento da capacidade de resposta, de eficácia e de desempenho na execução de tarefas e intervenções de manutenção no sítio e/ou por intermédio de aplicações remotas, dando um conjunto apropriado de informação relevante para a execução rápida e eficaz dessas tarefas, evitando assim a indisponibilidade dos sistemas por longos períodos de tempo.

Por outro lado, este sistema inteligente de gestão pode apresentar alguns condicionalismos tais como:

- ✓ Lidar com grandes quantidades de informação dependo do número de recursos e equipamentos envolvidos.

- ✓ A inexistência de um conjunto de regras pré-definidas para formação de equipas de manutenção pode levantar algumas discrepâncias na formação de equipas de manutenção, provocando solicitações repetidas de operadores mais experientes.
- ✓ Esta plataforma está sujeita aos condicionalismos de comunicação, impedindo o auxílio e apoio nas intervenções de manutenção remotas.

Resumindo, esta aplicação inclui portanto a implementação e comunicação entre 3 principais módulos da plataforma apresentada (módulo de alarme, módulo MTC e módulo MOSM) e a aplicação de uma plataforma de integração de Web Services com sistemas multiagentes **JADE**, faltando a implementação dos restantes módulos apresentados na arquitectura de referência apresentada.

Por fim, a não concretização da aplicação prática deste sistema inteligente não permitiu concluir o impacto, nem determinar indicadores de sucesso quanto ao desempenho e às melhorias pretendidas na integração deste sistema no contexto industrial, no entanto esta arquitectura de referência mereceu o destaque e a aprovação na comunidade científica resultando na publicação de dois artigos (Ribeiro, Barata et al. 2008; Ribeiro, Barata et al. 2009).

6.2 Perspectivas de Trabalho Futuro

Face à complexidade do sistema apresentado e por constrangimentos de tempo, a arquitectura foi implementada como uma prova de conceito através da simulação de um caso de teste.

Neste sentido, este sistema deve ser aplicado e testado adicionando equipamentos industriais, recursos e informação útil para o funcionamento em casos reais.

Por outro lado, deve ser efectuado um trabalho de melhoria na implementação dos módulos já existentes, adicionando funcionalidades importantes para a autonomia e capacidade de decisão do sistema, aplicando algoritmos e técnicas de decisão apropriados.

A pesquisa e implementação de novos componentes não existentes na arquitectura podem ser avaliadas e adicionadas como um complemento, de modo a aumentar o leque de funcionalidades no sistema de Manutenção.

Também à medida que o volume de informação e número de utilizadores do sistema inteligente aumente, devem ser substituídos o sistema e a base de conhecimento em documentos **XML** para uma aplicação de base de dados permitindo níveis de resposta, eficiência e desempenho superiores.

7 Referências Bibliográficas

- . "FIPA - The Foundation for Intelligent Physical Agents." from <http://www.fipa.org>.
- . "JADE (Java Agent DEvelopment Framework)." from <http://jade.cselt.it/>.
- Amari, S. V. (2006). "Bounds on MTBF of Systems Subjected to Periodic Maintenance." Reliability, IEEE Transactions on **55**(3): 469-474.
- Bangemann, T., X. Rebeuf, et al. (2006). "PROTEUS--Creating distributed maintenance systems through an integration platform." Computers in Industry **57**(6): 539-551.
- Bangemann, T., J. P. Thomesse, et al. (2004). PROTEUS - providing a concept for integrating online data into global maintenance strategies. Industrial Informatics, 2004. INDIN '04. 2004 2nd IEEE International Conference on.
- Bellifemine, F., G. Caire, et al. (2003). "JADE: A White Paper." EXP in search of innovation **3**(3): 6-19.
- Bellifemine, F. L., G. Caire, et al. (2007). Developing Multi-Agent Systems with JADE. Sussex, England.
- Benoit, I. and M. Adolfo Crespo (2006). "Special issue on e-maintenance." Comput. Ind. **57**(6): 473-475.
- Booth, D., H. Haas, et al. (2004). "Web Services Architecture."
- Box, D., D. Ehnebuske, et al. (2000). "Simple Object Access Protocol (SOAP) 1.1 ", from <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>.
- Bray, T., J. Paoli, et al. (1998). "Extensible Markup Language (XML) 1.0." from <http://www.w3.org/TR/1998/REC-xml-19980210.pdf>.
- Bukowski, J. V. and W. M. Goble (2001). "Defining mean time-to-failure in a particular failure-state for multi-failure-state systems." Reliability, IEEE Transactions on **50**(2): 221-228.
- Christensen, E., F. Curbera, et al. (2001). "Web Services Description Language (WSDL) 1.1." from <http://www.w3.org/TR/wsdl>.
- Clement, L., A. Hately, et al. (2004). "UDDI Version 3.0.2." from http://www.uddi.org/pubs/uddi_v3.htm#_Toc85907977.
- Dhillon, B. S. (2006). Maintainability, Maintenance, and Reliability for Engineers, CRC Press.
- Fallside, D. C. and P. Walmsley. (2004). "XML Schema Part 0: Primer Second Edition." from <http://www.w3.org/TR/xmlschema-0/>.
- FIPA. (2002). "FIPA Contract Net Interaction Protocol Specification (SC00029H)." from <http://www.fipa.org/specs/fipa00029/SC00029H.html>.
- FIPA. (2002). "FIPA Request Interaction Protocol Specification (SC00026)." from <http://www.fipa.org/specs/fipa00026/SC00026H.html>.

- Garcia, E., H. Guyennet, et al. (2004). "A new industrial cooperative tele-maintenance platform." Computers & Industrial Engineering **46**(4): 851-864.
- Garg, A. and S. G. Deshmukh (2006). "Maintenance management: literature review and directions." Journal of Quality in Maintenance Engineering **12**: 205-238.
- Han, T. and B.-S. Yang (2006). "Development of an e-maintenance system integrating advanced techniques." Computers in Industry **57**(6): 569-580.
- Huang, R., L. Xi, et al. (2005). "The framework, impact and commercial prospects of a new predictive maintenance system: intelligent maintenance system." Production Planning & Control: The Management of Operations **16**(7): 652 - 664.
- IMS. (2007). "Center for Intelligent Maintenance System." from <http://www.imscenter.net/>.
- JADE-Board (2008). "JADE Web Services Integration Gateway (WSIG) Guide." **Version 2**.
- Jay, L., N. Jun, et al. (2006). "Intelligent prognostics tools and e-maintenance." Comput. Ind. **57**(6): 476-489.
- Koç, M., J. Ni, et al. (2003). "Introduction to e-Manufacturing." International Journal of Agile Manufacturing **Vol. 6**(Special Issue on Distributed E-Manufacturing).
- Lee, J. (2003). "E-manufacturing--fundamental, tools, and transformation." Robotics and Computer-Integrated Manufacturing **19**(6): 501-507.
- Luck, M., P. McBurney, et al. (2005). Agent Technology: Computing as Interaction (A Roadmap for Agent Based Computing), AgentLink.
- Luck, M. a. M., P. and Shehory, O. and Willmott, S (2005). Agent Technology: Computing as Interaction (A Roadmap for Agent Based Computing), AgentLink.
- Marquez, A. C. (2007). The E-maintenance Revolution. The Maintenance Management Framework, Springer London: 305-327.
- Mobley, R. K. (2002). Impact of Maintenance. An Introduction to Predictive Maintenance (Second Edition), Butterworth-Heinemann.
- Mobley, R. K., L. R. Higgins, et al. (2008). Maintenance Engineering Handbook, McGraw-Hill.
- Mondro, M. J. (2002). "Approximation of mean time between failure when a system has periodic maintenance." Reliability, IEEE Transactions on **51**(2): 166-167.
- Monostori, L., J. Váncza, et al. (2006). "Agent-Based Systems for Manufacturing." CIRP Annals - Manufacturing Technology **55**(2): 697-720.
- Moubray, J. (1997). Reliability-centered maintenance, Industrial Press Inc.
- Muller, A., A. Crespo Marquez, et al. (2008). "On the concept of e-maintenance: Review and current research." Reliability Engineering & System Safety **93**(8): 1165-1187.
- Ribeiro, L., J. Barata, et al. (2009). Maintenance Management and Operational Support as Services in Reconfigurable Manufacturing Systems. IFAC Symposium on INFORMATION CONTROL PROBLEMS IN MANUFACTURING. Moscow, IFAC.
- Ribeiro, L., J. Barata, et al. (2008). "A high level e-maintenance architecture to support on-site teams." Enterprise and Work Innovation Studies **4**(4): 129-138.
- Szymanski, J., T. Bangemann, et al. (2003). PROTEUS - a European initiative for e-maintenance platform development. Emerging Technologies and Factory Automation, 2003. Proceedings. ETFA '03. IEEE Conference.
- Takata, S., F. Kirnura, et al. (2004). "Maintenance: Changing Role in Life Cycle Management." CIRP Annals - Manufacturing Technology **53**(2): 643-655.
- Tang, H., E. Miller-Hooks, et al. (2007). "Scheduling technicians for planned maintenance of geographically distributed equipment." Transportation Research Part E: Logistics and Transportation Review **43**(5): 591-609.
- Tsang, A. H. C. (2002). "Strategic dimensions of maintenance management." Journal of Quality in Maintenance Engineering **8**: 7-39.

- Yam, R. C. M., P. W. Tse, et al. (2001). "Intelligent Predictive Decision Support System for Condition-Based Maintenance." The International Journal of Advanced Manufacturing Technology **17**(5): 383-391.
- Yang, S. H., C. Dai, et al. (2007). "Remote maintenance of control system performance over the Internet." Control Engineering Practice **15**(5): 533-544.
- Yu, R., B. Iung, et al. (2003). "A multi-agents based E-maintenance system with case-based reasoning decision support." Engineering Applications of Artificial Intelligence **16**(4): 321-333.

Anexo I – Web Service de Manutenção

```
<?xml version="1.0" encoding="UTF-8"?>
  <wsdl:definitions name="maintenanceWSService"
    targetNamespace="urn:maintenanceWSService"
    xmlns:impl="urn:maintenanceWSService"
    xmlns:wsdlsoap=http://schemas.xmlsoap.org/wsdl/soap/
    xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
    <wsdl:types>
      <xsd:schema targetNamespace="urn:maintenanceWSService"
        xmlns:impl="urn:maintenanceWSService"
        xmlns:xsd="http://www.w3.org/2001/XMLSchema"><xsd:annotation/>
        <xsd:element name="deviceunregistration">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="uniqueId" type="xsd:string"/>
              <xsd:element name="serialNumber" type="xsd:string"/>
              <xsd:element name="portNumber" type="xsd:string"/>
              <xsd:element name="timeStamp" type="xsd:string"/>
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="deviceunregistrationResponse">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="deviceunregistrationReturn"
                type="impl:unregisterReturn"/>
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
        <xsd:complexType name="unregisterReturn">
          <xsd:sequence>
            <xsd:element name="status" type="impl:status"/>
          </xsd:sequence>
        </xsd:complexType>
        <xsd:complexType name="status">
          <xsd:sequence>
            <xsd:element name="type" type="xsd:int"/>
            <xsd:element name="message" type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
        <xsd:element name="deviceregistration">
          <xsd:complexType>
            <xsd:sequence>
```

```

        <xsd:element name="name" type="xsd:string"/>
        <xsd:element minOccurs="0" name="friendlyName"
            type="xsd:string"/>
        <xsd:element name="serialNumber" type="xsd:string"/>
        <xsd:element name="portNumber" type="xsd:string"/>
        <xsd:element name="type" type="xsd:string"/>
        <xsd:element minOccurs="0" name="description"
            type="xsd:string"/>
        <xsd:element minOccurs="0" name="associatedSystemList"
            type="impl:ArrayOfString"/>
        <xsd:element minOccurs="0" name="faultsList"
            type="impl:ArrayOfFault"/>
        <xsd:element name="heartBeatFrequency" type="xsd:int"/>
        <xsd:element name="timeStamp" type="xsd:string"/>
    </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:complexType name="ArrayOfString">
    <xsd:sequence>
        <xsd:element maxOccurs="unbounded" minOccurs="0"
            name="string" type="xsd:string"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ArrayOfFault">
    <xsd:sequence>
        <xsd:element maxOccurs="unbounded" minOccurs="0"
            name="fault" type="impl:fault"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="fault">
    <xsd:sequence>
        <xsd:element name="faultId" type="xsd:string"/>
        <xsd:element name="faultDescription" type="xsd:string"/>
        <xsd:element name="faultType" type="xsd:string"/>
        <xsd:element name="timeStamp" type="xsd:string"/>
        <xsd:element name="sourceId" type="xsd:string"/>
        <xsd:element name="serialNumber" type="xsd:string"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:element name="deviceregistrationResponse">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="deviceregistrationReturn"
                type="impl:registerReturn"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:complexType name="registerReturn">
    <xsd:sequence>
        <xsd:element name="uniqueId" type="xsd:string"/>
        <xsd:element name="serialNumber" type="xsd:string"/>
        <xsd:element name="portNumber" type="xsd:string"/>
        <xsd:element name="status" type="impl:status"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:element name="maintenancereport">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="uniqueId" type="xsd:string"/>
            <xsd:element name="messageType" type="xsd:int"/>
            <xsd:element minOccurs="0" name="description"
                type="xsd:string"/>
            <xsd:element minOccurs="0" name="severity"
                type="xsd:int"/>

```

```

        <xsd:element minOccurs="0" name="uptime"
            type="xsd:int" />
        <xsd:element name="maintenanceCode"
            type="xsd:string" />
        <xsd:element minOccurs="0"
            name="maintenanceDescription"
            type="xsd:string" />
        <xsd:element minOccurs="0" name="mtbf"
            type="xsd:string" />
        <xsd:element minOccurs="0" name="mttr"
            type="xsd:string" />
        <xsd:element name="timeStamp" type="xsd:string" />
    </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="maintenancereportResponse">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="maintenancereportReturn"
                type="impl:reportingMaintenanceReturn" />
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:complexType name="reportingMaintenanceReturn">
    <xsd:sequence>
        <xsd:element name="uniqueId" type="xsd:string" />
        <xsd:element name="status" type="impl:status" />
    </xsd:sequence>
</xsd:complexType>
<xsd:element name="diagnosisreport">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="uniqueId" type="xsd:string" />
            <xsd:element name="messageType" type="xsd:int" />
            <xsd:element minOccurs="0" name="description"
                type="xsd:string" />
            <xsd:element minOccurs="0" name="severity"
                type="xsd:int" />
            <xsd:element minOccurs="0" name="uptime"
                type="xsd:int" />
            <xsd:element minOccurs="0" name="faultsHistory"
                type="impl:ArrayOfFault" />
            <xsd:element minOccurs="0" name="currentFaults"
                type="impl:ArrayOfFault" />
            <xsd:element name="diagnosticCode"
                type="xsd:string" />
            <xsd:element name="diagnosticDescription"
                type="xsd:string" />
            <xsd:element minOccurs="0" name="timeStamp"
                type="xsd:string" />
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="diagnosisreportResponse">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="diagnosisreportReturn"
                type="impl:reportingDiagnosisReturn" />
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:complexType name="reportingDiagnosisReturn">
    <xsd:sequence>
        <xsd:element name="uniqueId" type="xsd:string" />

```

```

        <xsd:element name="status" type="impl:status"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:element name="prognosisreport">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="uniqueId"
                type="xsd:string"/>
            <xsd:element name="messageType"
                type="xsd:int"/>
            <xsd:element minOccurs="0" name="description"
                type="xsd:string"/>
            <xsd:element minOccurs="0" name="severity"
                type="xsd:int"/>
            <xsd:element minOccurs="0" name="uptime"
                type="xsd:int"/>
            <xsd:element name="prognosticCode"
                type="xsd:string"/>
            <xsd:element minOccurs="0"
                name="prognosticDescription"
                type="xsd:string"/>
            <xsd:element name="timeStamp"
                type="xsd:string"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="prognosisreportResponse">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="prognosisreportReturn"
                type="impl:reportingPrognosisReturn"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:complexType name="reportingPrognosisReturn">
    <xsd:sequence>
        <xsd:element name="uniqueId"
            type="xsd:string"/>
        <xsd:element name="status"
            type="impl:status"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:element name="heartBeat">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="uniqueId"
                type="xsd:string"/>
            <xsd:element name="deviceStatus"
                type="xsd:int"/>
            <xsd:element name="timeStamp"
                type="xsd:string"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="heartBeatResponse">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="heartBeatReturn"
                type="impl:heartBeatReturn"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:complexType name="heartBeatReturn">
    <xsd:sequence>

```

```

        <xsd:element name="uniqueId" type="xsd:string"/>
        <xsd:element name="status" type="impl:status"/>
    </xsd:sequence>
</xsd:complexType>
</xsd:schema>
</wsdl:types>
<wsdl:message name="diagnosisreportResponse">
    <wsdl:part name="parameters" element="impl:diagnosisreportResponse">
    </wsdl:part>
</wsdl:message>
<wsdl:message name="deviceregistrationRequest">
    <wsdl:part name="parameters" element="impl:deviceregistration">
    </wsdl:part>
</wsdl:message>
<wsdl:message name="heartBeatRequest">
    <wsdl:part name="parameters" element="impl:heartBeat">
    </wsdl:part>
</wsdl:message>
<wsdl:message name="deviceunregistrationRequest">
    <wsdl:part name="parameters" element="impl:deviceunregistration">
    </wsdl:part>
</wsdl:message>
<wsdl:message name="prognosisreportRequest">
    <wsdl:part name="parameters" element="impl:prognosisreport">
    </wsdl:part>
</wsdl:message>
<wsdl:message name="heartBeatResponse">
    <wsdl:part name="parameters" element="impl:heartBeatResponse">
    </wsdl:part>
</wsdl:message>
<wsdl:message name="maintenancereportResponse">
    <wsdl:part name="parameters" element="impl:maintenancereportResponse">
    </wsdl:part>
</wsdl:message>
<wsdl:message name="deviceregistrationResponse">
    <wsdl:part name="parameters" element="impl:deviceregistrationResponse">
    </wsdl:part>
</wsdl:message>
<wsdl:message name="diagnosisreportRequest">
    <wsdl:part name="parameters" element="impl:diagnosisreport">
    </wsdl:part>
</wsdl:message>
<wsdl:message name="deviceunregistrationResponse">
    <wsdl:part name="parameters"
        element="impl:deviceunregistrationResponse">
    </wsdl:part>
</wsdl:message>
<wsdl:message name="prognosisreportResponse">
    <wsdl:part name="parameters" element="impl:prognosisreportResponse">
    </wsdl:part>
</wsdl:message>
<wsdl:message name="maintenancereportRequest">
    <wsdl:part name="parameters" element="impl:maintenancereport">
    </wsdl:part>
</wsdl:message>
<wsdl:portType name="maintenanceWSServicePort">
    <wsdl:operation name="deviceunregistration">
        <wsdl:input message="impl:deviceunregistrationRequest">
        </wsdl:input>
        <wsdl:output message="impl:deviceunregistrationResponse">
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="deviceregistration">
        <wsdl:input message="impl:deviceregistrationRequest">

```

```

</wsdl:input>
  <wsdl:output message="impl:deviceregistrationResponse">
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="maintenancereport">
  <wsdl:input message="impl:maintenancereportRequest">
</wsdl:input>
  <wsdl:output message="impl:maintenancereportResponse">
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="diagnosisreport">
  <wsdl:input message="impl:diagnosisreportRequest">
</wsdl:input>
  <wsdl:output message="impl:diagnosisreportResponse">
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="prognosisreport">
  <wsdl:input message="impl:prognosisreportRequest">
</wsdl:input>
  <wsdl:output message="impl:prognosisreportResponse">
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="heartBeat">
  <wsdl:input message="impl:heartBeatRequest">
</wsdl:input>
  <wsdl:output message="impl:heartBeatResponse">
</wsdl:output>
</wsdl:operation>
</wsdl:portType>
<wsdl:binding name="maintenanceWSServiceBinding"
type="impl:maintenanceWSServicePort">
  <wsdlsoap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
  <wsdl:operation name="deviceunregistration">
    <wsdlsoap:operation soapAction="urn:maintenanceWSServiceAction"/>
    <wsdl:input>
      <wsdlsoap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <wsdlsoap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="deviceregistration">
    <wsdlsoap:operation soapAction="urn:maintenanceWSServiceAction"/>
    <wsdl:input>
      <wsdlsoap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <wsdlsoap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="maintenancereport">
    <wsdlsoap:operation soapAction="urn:maintenanceWSServiceAction"/>
    <wsdl:input>
      <wsdlsoap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <wsdlsoap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="diagnosisreport">
    <wsdlsoap:operation soapAction="urn:maintenanceWSServiceAction"/>
    <wsdl:input>
      <wsdlsoap:body use="literal"/>

```

```

        </wsdl:input>
        <wsdl:output>
            <wsdlsoap:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="prognosisreport">
        <wsdlsoap:operation soapAction="urn:maintenanceWSServiceAction"/>
        <wsdl:input>
            <wsdlsoap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <wsdlsoap:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="heartBeat">
        <wsdlsoap:operation soapAction="urn:maintenanceWSServiceAction"/>
        <wsdl:input>
            <wsdlsoap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <wsdlsoap:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
</wsdl:binding>
<wsdl:service name="maintenanceWSServiceService">
    <wsdl:port name="maintenanceWSServicePort"
binding="impl:maintenanceWSServiceBinding">
        <wsdlsoap:address
location="http://192.168.1.106:8080/wsig/ws/maintenanceWSService"/>
    </wsdl:port>
</wsdl:service>
</wsdl:definitions>

```

Anexo II – Schemas

✓ *Schema deviceRegistrationBD*

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema id="deviceRegistrationBD"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.fct.unl.pt"
  xmlns="http://www.fct.unl.pt"
  elementFormDefault="qualified">

  <xsd:element name="RegistrationBD">
    <xsd:complexType>
      <xsd:sequence minOccurs="0" maxOccurs="unbounded">
        <xsd:element name="DeviceRegistration"
          type="DeviceRegister"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="DeviceRegister">
    <xsd:sequence>
      <xsd:element name="Register" type="DRegister"/>
      <xsd:element name="DeviceStatus" type="Status"/>
    </xsd:sequence>
    <xsd:attribute name="idRegister" type="xsd:string" use="required"/>
  </xsd:complexType>

  <xsd:complexType name="DRegister">
    <xsd:sequence>
      <xsd:element name="name" type="xsd:string"/>
      <xsd:element name="friendlyName" type="xsd:string"/>
      <xsd:element name="serialNumber" type="xsd:string"/>
      <xsd:element name="portNumber" type="xsd:string"/>
      <xsd:element name="type" type="xsd:integer"/>
      <xsd:element name="description" type="xsd:string"/>
      <xsd:element name="faultsList" type="FaultsList"/>
      <xsd:element name="associatedSystemList"
        type="AssociatedSystemList"/>
      <xsd:element name="heartBeatFrequency" type="xsd:integer"/>
      <xsd:element name="timeStamp" type="xsd:string"/>
    </xsd:sequence>
  </xsd:complexType>
```



```

<xsd:complexType name="AssociatedSystemList">
  <xsd:sequence minOccurs="0" maxOccurs="unbounded">
    <xsd:element name="associatedSystem" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="FaultsList">
  <xsd:sequence minOccurs="0" maxOccurs="unbounded">
    <xsd:element name="fault" type="Fault"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="Fault">
  <xsd:sequence>
    <xsd:element name="faultId" type="xsd:string"/>
    <xsd:element name="faultDescription" type="xsd:string"/>
    <xsd:element name="faultType" type="xsd:string"/>
    <xsd:element name="timeStamp" type="xsd:string"/>
    <xsd:element name="sourceId" type="xsd:string"/>
    <xsd:element name="serialNumber" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="Status">
  <xsd:sequence>
    <xsd:element name="statusType" type="StatusType"/>
    <xsd:element name="message" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name="StatusType">
  <xsd:restriction base="xsd:integer">
    <xsd:enumeration value="0"/>
    <xsd:enumeration value="1"/>
  </xsd:restriction>
</xsd:simpleType>

</xsd:schema>

```

✓ *Schema maintenanceTeamBD*

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema id="MaintenanceTeamBD"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.fct.unl.pt"
  xmlns="http://www.fct.unl.pt"
  elementFormDefault="qualified">

  <xsd:complexType name="HistoryOfOperations">
    <xsd:sequence minOccurs="0" maxOccurs="unbounded">
      <xsd:element name="MaintenanceOperations"
        type="MaintenanceOperation"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="MaintenanceOperation">
    <xsd:sequence>
      <xsd:element name="name" type="xsd:string"/>
      <xsd:element name="type" type="xsd:string"/>
      <xsd:element name="idDevice" type="xsd:string"/>
      <xsd:element name="statusMaintenanceOperation"
        type="StatusOperation"/>
    </xsd:sequence>
  </xsd:complexType>

```

```

        <xsd:element name="reason" type="xsd:string"/>
        <xsd:element name="startDate" type="xsd:dateTime" />
        <xsd:element name="pendingDate" type="xsd:dateTime"/>
        <xsd:element name="endDate" type="xsd:dateTime" nillable = "true"/>
        <xsd:element name="cost" type="xsd:decimal"/>
    </xsd:sequence>
    <xsd:attribute name="idOperation" type="xsd:string" use="required"/>
</xsd:complexType>

<xsd:simpleType name="StatusOperation">
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value="Started"/>
        <xsd:enumeration value="Cancelled"/>
        <xsd:enumeration value="Suspended"/>
        <xsd:enumeration value="Finished"/>
    </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="Skills">
    <xsd:sequence minOccurs="0" maxOccurs="unbounded">
        <xsd:element name="MaintenanceSkill" type="Skill"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="Skill">
    <xsd:sequence>
        <xsd:element name="name" type="xsd:string"/>
        <xsd:element name="value" type="xsd:string"/>
    </xsd:sequence>
    <xsd:attribute name="idSkill" type="xsd:string" use="required"/>
</xsd:complexType>

<xsd:complexType name="Employee">
    <xsd:sequence>
        <xsd:element name="name" type="xsd:string" />
        <xsd:element name="email" type="xsd:string" />
        <xsd:element name="valueOfEmployee" type="xsd:decimal"/>
        <xsd:element name="status" type="StatusEmployee"/>
        <xsd:element name="MaintenanceSkills" type="Skills"/>
        <xsd:element name="History" type="HistoryOfOperations"/>
    </xsd:sequence>
    <xsd:attribute name="idEmployee" type="xsd:string" use="required"/>
</xsd:complexType>

<xsd:simpleType name="StatusEmployee">
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value="Allocated"/>
        <xsd:enumeration value="Available"/>
        <xsd:enumeration value="Unavailable"/>
    </xsd:restriction>
</xsd:simpleType>

<xsd:element name="MaintenanceTeam">
    <xsd:complexType>
        <xsd:sequence minOccurs="0" maxOccurs="unbounded">
            <xsd:element name="MaintenanceEmployee" type="Employee"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
</xsd:schema>

```

✓ *Schema deviceSpecificationBD*

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema id="DeviceSpecificationBD"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.fct.unl.pt"
  xmlns="http://www.fct.unl.pt"
  elementFormDefault="qualified">

  <xsd:element name="DeviceSpecificationRepository">
    <xsd:complexType>
      <xsd:sequence minOccurs="0" maxOccurs="unbounded">
        <xsd:element name="DeviceSpecification"
          type="DeviceSpecification"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="DeviceSpecification">
    <xsd:sequence>
      <xsd:element name="DeviceDescription" type="Description"/>
      <xsd:element name="GlobalSpecification"
        type="Specification"/>
      <xsd:element name="MaintenanceSpecification"
        type="MaintenanceSpecification"/>
    </xsd:sequence>
    <xsd:attribute name="typeDevice" type="xsd:string" use="required"/>
  </xsd:complexType>

  <xsd:complexType name="Description">
    <xsd:sequence>
      <xsd:element name="name" type="xsd:string"/>
      <xsd:element name="model" type="xsd:string"/>
      <xsd:element name="family" type="xsd:string"/>
      <xsd:element name="serialNumber" type="xsd:string"/>
      <xsd:element name="portNumber" type="xsd:string"/>
      <xsd:element name="manufacturer" type="xsd:string"/>
      <xsd:element name="manufacturerAddress" type="xsd:string"/>
      <xsd:element name="manufacturerContact" type="xsd:string"/>
      <xsd:element name="dateConstruction" type="xsd:dateTime"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="Specification">
    <xsd:sequence>
      <xsd:element name="ElectricalProperties"
        type="ElectricalProperties" nillable="true"/>
      <xsd:element name="PerformanceProperties"
        type="PerformanceProperties" nillable="true"/>
      <xsd:element name="EnvironmentProperties"
        type="EnvironmentProperties" nillable="true"/>
      <xsd:element name="PneumaticProperties"
        type="PneumaticProperties" nillable="true"/>
      <xsd:element name="PhysicalProperties"
        type="PhysicalProperties" nillable="true"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="PhysicalProperties">
    <xsd:sequence>
      <xsd:element name="weight" type="xsd:decimal" nillable="true"/>
      <xsd:element name="height" type="xsd:decimal" nillable="true"/>
      <xsd:element name="MomentOfInertia" type="MomentOfInertia"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

```

        nillable="true"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="MomentOfInertia">
    <xsd:sequence>
        <xsd:element name="momentInertia_x" type="xsd:decimal"
            nillable="true"/>
        <xsd:element name="momentInertia_y" type="xsd:decimal"
            nillable="true"/>
        <xsd:element name="momentInertia_z" type="xsd:decimal"
            nillable="true"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="PneumaticProperties">
    <xsd:sequence>
        <xsd:element name="pneumaticPressure" type="RangeValues"
            nillable="true"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="ElectricalProperties">
    <xsd:sequence>
        <xsd:element name="supplyVoltage" type="RangeValues"
            nillable="true"/>
        <xsd:element name="supplyFrequency" type="RangeValues"
            nillable="true"/>
        <xsd:element name="supplyTransformer" type="RangeValues"
            nillable="true"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="PerformanceProperties">
    <xsd:sequence>
        <xsd:element name="positionRepeatability" type="xsd:string"
            nillable="true"/>
        <xsd:element name="responseTime" type="xsd:string"
            nillable="true"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="EnvironmentProperties">
    <xsd:sequence>
        <xsd:element name="ambientTemperature" type="RangeValues"
            nillable="true"/>
        <xsd:element name="relativeHumidity" type="RangeValues"
            nillable="true"/>
        <xsd:element name="noiseLevel" type="xsd:string"
            nillable="true"/>
        <xsd:element name="safety" type="xsd:string" nillable="true"/>
        <xsd:element name="emmission" type="xsd:string"
            nillable="true"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="MaintenanceSpecification">
    <xsd:sequence>
        <xsd:element name="mttf" type="xsd:integer" nillable="true"/>
        <xsd:element name="mtbf" type="xsd:integer" nillable="true"/>
        <xsd:element name="mttr" type="xsd:integer" nillable="true"/>
        <xsd:element name="uptime" type="xsd:integer" nillable="true"/>
        <xsd:element name="idleTime" type="xsd:integer"
            nillable="true"/>
    </xsd:sequence>
</xsd:complexType>

```

```

        <xsd:element name="unhandledEvents" type="xsd:integer"
                    nillable="true"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="RangeValues">
    <xsd:sequence>
        <xsd:element name="minimumValue" type="xsd:string"
                    nillable="true"/>
        <xsd:element name="maximumValue" type="xsd:string"
                    nillable="true"/>
        <xsd:element name="suggestedValue" type="xsd:string"
                    nillable="true"/>
    </xsd:sequence>
</xsd:complexType>
</xsd:schema>

```

✓ *Schema errorMapping*

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema id="ErrorMappingBD"
            xmlns:xsd="http://www.w3.org/2001/XMLSchema"
            targetNamespace="http://www.fct.unl.pt"
            xmlns="http://www.fct.unl.pt"
            elementFormDefault="qualified">

    <xsd:element name="ErrorMapping">
        <xsd:complexType>
            <xsd:sequence minOccurs="0" maxOccurs="3">
                <xsd:element name="map" type="Mapping"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>

    <xsd:complexType name="Mapping">
        <xsd:sequence minOccurs="0" maxOccurs="unbounded">
            <xsd:element name="errorHandling" type="ErrorMap"/>
        </xsd:sequence>
        <xsd:attribute name="type" type="xsd:string" use="required"/>
    </xsd:complexType>

    <xsd:complexType name="ErrorMap">
        <xsd:sequence>
            <xsd:element name="description" type="xsd:string"/>
            <xsd:element name="solutions" type="ErrorSolution"/>
        </xsd:sequence>
        <xsd:attribute name="code" type="xsd:string" use="required"/>
    </xsd:complexType>

    <xsd:complexType name="ErrorSolution">
        <xsd:sequence minOccurs="0" maxOccurs="unbounded">
            <xsd:element name="solution" type="xsd:string"/>
        </xsd:sequence>
    </xsd:complexType>

</xsd:schema>

```